AFRL-SR-BL-TR-98-

C460

REPORT DOCUMENTATION PAGE

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	03/01/98	Progress Report 01 August 1996 - 31 July 1997 and Final Report 30 September 1995 - 29 November 1997

4. TITLE AND SUBTITLE

Multiscale Photonic Data Fusion Networks and Their Interfaces

5. FUNDING NUMBERS

6. AUTHORS

Alexander A. Sawchuk, P.I.

C.-C. Jay Kuo, Victor Li, and Alan Willner, Co-Investigators

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

University of Southern California

Signal and Image Processing Institute

3740 McClintock Avenue, Suite 404

Los Angeles, CA 90089-2564

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AFOSR/NE

110 Duncan Avenue, Suie B115

Bolling Air Force Base, DC 20332-0001

ATTN: Dr. Alan E. Craig, Program Officer

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

19980602 038

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release: distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

We describe results for the period 30 September 1995 - 29 November 1997 in a basic research program on Multiscale Photonic Data Fusion Networks and Their Interfaces, part of the DOD Focused Research Initiative (FRI) program performed at the University of Southern California (USC). The research results include: high speed photonic network architectures, photonic system design and integration for space, wavelength and time signal processing and transmission, and image processing, compression and coding algorithms. The report contents includes: Optoelectronic Interfaces for Optical Page-Oriented Memory and High Speed; Wavelength-Division-Multiplexing for High-Speed Network Gateways; Pruned Octree Feature for Interactive Retrieval.

14. SUBJECT TERMS

15. NUMBER OF PAGES

71

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNLIMITED

US DEPARTMENT OF DEFENSE
FOCUSED RESEARCH INITIATIVE (FRI)

Topic Area: Photonics for Data Fusion Networks

Multiscale Photonic Data Fusion Networks and Their Interfaces

Combined Report:
Progress Report for the Period 01 August 1996 – 31 July 1997
Final Technical Report for the Period 30 September 1995 – 29 November 1997

Submitted to:

Dr. Alan Craig
AFOSR/NE
Bolling AFB
Building 410
Washington, DC 20332-0001

March 1, 1998

Principal Investigator:

Alexander A. Sawchuk
Professor of Electrical Engineering
University of Southern California
Signal and Image Processing Institute
3740 McClintock Avenue, Room 404
Los Angeles, CA 90089-2564
Tel: 213-740-4622
Fax: 213-740-4651
Email: sawchuk@sipi.usc.edu

Nothing on this page is classified or proprietary information/data

1.0 ABSTRACT, BACKGROUND AND RESEARCH ISSUES

We describe results for the period 30 September 1995 - 29 November 1997 in a basic research program on Multiscale Photonic Data Fusion Networks and Their Interfaces, part of the DOD Focused Research Initiative (FRI) program performed at the University of Southern California (USC). The research results include: high speed photonic network architectures, photonic system design and integration for space, wavelength and time signal processing and transmission, and image processing, compression and coding algorithms. The report contents includes: Optoelectronic Interfaces for Optical Page-Oriented Memory and High Speed; Wavelength-Division-Multiplexing for High-Speed Network Gateways; Pruned Octree Feature for Interactive Retrieval.

2.0 Optoelectronic Interfaces for Optical Page-Oriented Memory and High Speed Networks, Alexander A. Sawchuk

2.1 Summary

In this report we present the design and analysis of an error correction coding/decoding interface for optical page-oriented memories. The interface utilizes smart pixel technology (SP) to provide high data access rates. The interface contains an array of SP Reed-Solomon (RS) decoders that implement the transfer decoding algorithm (TDA) to reduce the relatively high raw output BER of 10^{-4} to 10^{-7} to a BER of 10^{-12} or better. The TDA is implemented by 1-D and 2-D pipeline structures and serial and parallel finite field multipliers, resulting in six design variations. A modified VLSI circuit simulation model was employed to estimate the decoder area and power dissipation. Two analyses were performed: (1) defining system parameters for the RS coder and decoder which provide the highest aggregate output throughput of corrected information bits; (2) determining RS coder and decoder design which provide the highest code rate (fraction of corrected information bits out of all output bits) and, in turn, achieve the largest usable capacity. The results show that the codeword length of the best RS codes tends to approach two extremes: achieving either high data throughput (shorter length codes), or high capacity (longer length codes).

2.2 Introduction

Current digital processors have clock rates in excess of 200 MHz, a rate higher than the output rates of conventional secondary data storage technology. In addition to computer applications, enhanced digital information services, such as high-resolution multimedia images, video-on-demand, and high definition television, require the storage of a large amount of data at very low bit-error rates (BER), fast access to this data, and the efficient interface of the storage system to high speed, gigabit per second networks [1]. The information bandwidth of communication networks using optoelectronic integrated devices and optical-fiber transmission has increased rapidly to the current 2.4 Gbits/second/channel and to 10 Gbits/second/channel in near future [2]. The performance of information systems is inevitably limited by the access rate of the data storage systems. Optical page-oriented memory (OPOM) [3]-[5] technology is one candidate that simultaneously provides large capacity (10^{12} bits/cm^3 theoretically) and a high data access rate (10^9 bits/second or more). Unfortunately, a limitation of OPOMs is that they have a high raw BER (in the range of 10^{-4} to 10^{-7}). The use of error detection/correction is one way to reduce the BER to a desirable rate while maintaining the large overall memory capacity [1], [6].

Interfaces between OPOMs and high speed networks must not only provide high data throughput rates to prevent I/O bottlenecks, but must also reduce the BER. Because high performance error-correction encoding/decoding requires complicated operations and hardware, the overall data throughput may need to be reduced if extensive error correction is needed. Thus, there is a tradeoff between the data throughput and error-correction capability. This study focuses on the error correction decoding in the output interface of OPOMs as shown in Fig. 2.1. The decoding logic in the interface operates on the binary bit streams received from binary thresholding detection that is performed at the photodetector. Decoded binary data, whose BER is ideally 10^{-12} or better, is output to a network or computing system. While error encoding/decoding may be needed to process the data for its transmission in the network, this subject has been well studied in the field of digital data transmission and is outside the scope of this study.

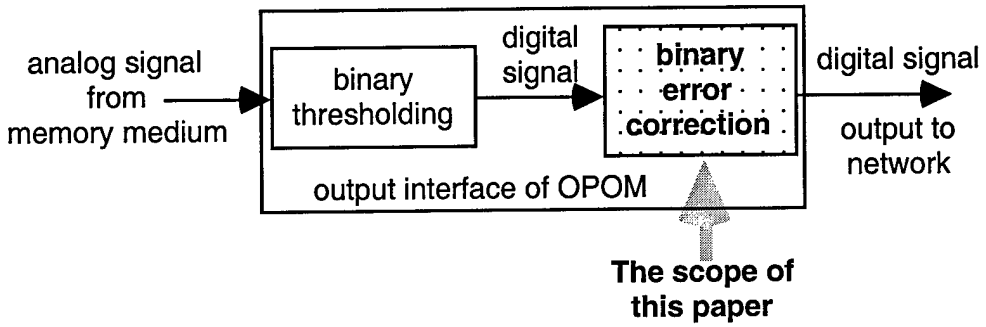

Figure 2.1 Scope of this work. OPOM denotes an optical page-oriented memory.

Error correcting codes such as Reed-Solomon (RS) codes have been successfully used to correct random and burst errors in deep space communications, and in data retrieval from mass storage devices and optical disks [7]. Because of the complicated decoding processes of high performance codes, however, the decoding rate is limited to several tens of megabits per second [8]. Neifeld proposed an RS decoder with parallel input/output (I/O) with an information rate of 300 megabits/second, and constructed a module with an array of parallel decoders [9], [10].

In order to provide gigabit-per-second aggregate decoding rate, multiple error-correcting decoders at the output of the OPOM systems are needed. Optoelectronic (OE) smart pixel (SP) devices are one means of achieving high data rates by parallel processing and a large number of I/O. The SP technology uses VLSI processes for monolithic integration of photodetectors for data input, optical modulators or sources for data output, and electrical circuitry for computing and logical operations [11]. The SP interface consists of an array of SP decoders, and each has electrical circuitry to perform decoding computations and logic; and optical I/O to transfer data to/from local pixels or other devices.

The goal of this study is to design output interfaces for a smart pixel OPOM system with high aggregate data rates (10^{11} bits/second or 0.1 terabits/second) and large usable capacity while reducing the BER to an acceptable rate (10^{-12} or better). Different design variations of RS decoders used in the SP interfaces were studied. The RS code and decoder design that provides the highest aggregate data rate and the largest usable memory capacity were determined under the limits of given physical conditions such as the minimum feature size of VLSI processes, chip area, and power density. The scenarios developed here may also be applied to evaluate the performance of other SP designs.

We have studied six RS decoder variations which are constructed from four decoder designs, with bitwise and/or symbolwise I/O formats, devised with two types of finite field multipliers. All these designs employ an RS decoding scheme, the transform decoding algorithm, which has a regular structure and is suitable for VLSI implementation [12]. Since there are six implementations and each has many choices of possible RS codes, an objective is to determine which implementation and RS code provides the highest data throughput under the limits of a fixed VLSI minimum feature size, chip area, and power density. For an OPOM, on the other hand, the minimum access time and the size of a data page are determined by the material characteristics. The corresponding addressing scheme is known in advance, so the data rate entering the output interface is determined. Designers must choose the RS code and the decoder implementation which are able to best match the established data rate while achieving a large usable capacity.

This report contains five major sections. Section 2.3 describes the fundamentals of the optical page-oriented memories and summarizes current smart pixel technology. Section 2.4 reviews error-correction codes, particularly, Reed-Solomon codes that are adopted here. Section 2.5

describes smart pixel interfaces and compares various designs. Section 2.6 shows the performance of the decoder variations and the feasibility analysis of the SP interfaces using the designed decoders. Section 2.7 concludes and summarizes key results. Appendix A describes a VLSI circuit simulation model, modified SUSPENS, which is used to estimate the decoder area, power dissipation, and maximum clock frequency.

2.3 Preliminaries

2.3.1 Optical Page-Oriented Memories

Optical page-oriented memory (OPOM) [1], [3]-[5] has potentially large storage capacity, short access time, and high data access rate to satisfy the requirements of advanced audio, video, and multimedia applications. Recent developments in materials, spatial light modulators, and solid-state lasers have revitalized the OPOM, which was first proposed in 1963 [13]. An OPOM consists of an input interface, a memory medium, and an output interface, as shown in Fig. 2.2 [9]. Note that only recording material and I/O interfaces are shown; however, in a complete memory system, an addressing system will be included as well. In this section, we present the characteristics of these modules.

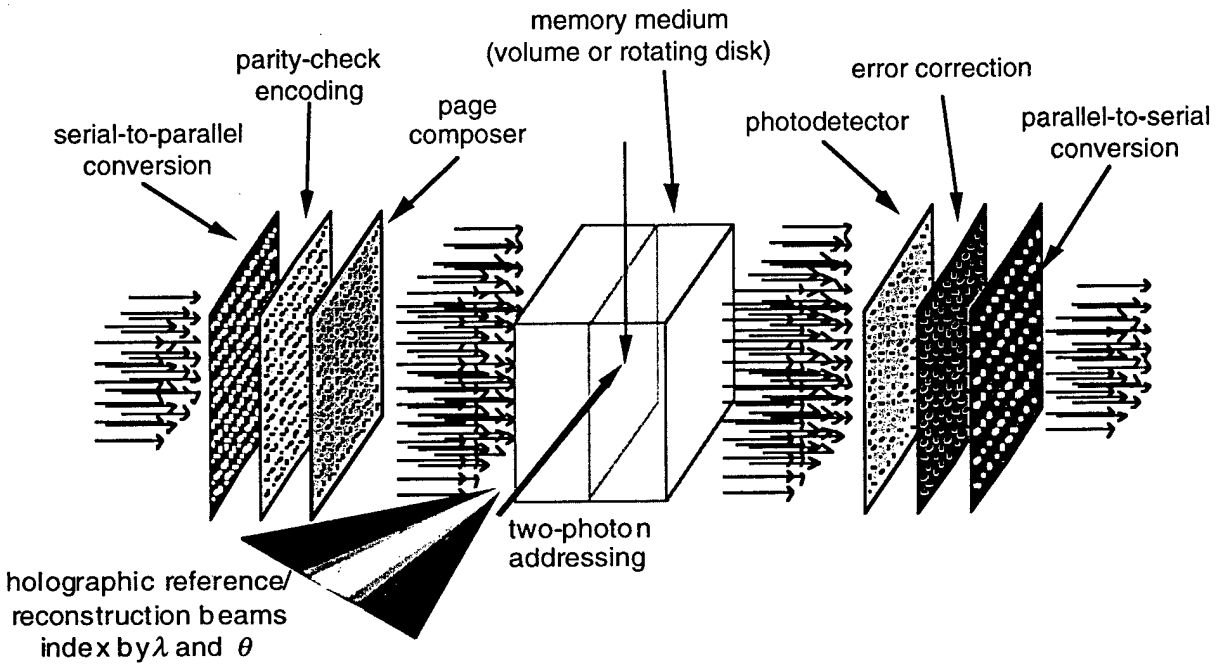

Figure 2.2 Optical page-oriented memory and input/output interfaces.

Input Interface

The input interface consists of a serial-to-parallel converter, an error-correction parity-check encoder, and a page composer. The serial-to-parallel converter may be composed of an electrical interface or optical receivers such as a CCD or photodetector array along with signal buffers. The error-correcting encoder is needed when the data to be stored has not previously encoded. In many applications (such as archival data storage), the data input operation takes place off-line at a relatively slow speed compared to readout. The page composer formats the data to be input in the memory. The page composer consists of many independent spatial elements whose transmittance is modulated by a large number of electrical or optical data channels. Data may then be read out in parallel by a light beam, and each array of data is called a page. Liquid-crystal spatial light modulators (SLMs) and film masks are examples of SLMs usually used for dynamic and static data input, respectively. Resolution and fram time are major issues for the page composer. Page

composers with more data bits per data page and faster frame time can achieve larger capacity and higher data rates. The number of data bits is also limited, however, by the imaging system, noise effects of the memory medium, and the capacity of the output devices. The size of a data page may be on the order of a million bits.

Memory Medium and the Recording/Retrieving Methods

There are several materials and technologies that have been developed for the OPOM. These include: photorefractive crystals; two-photon materials; spectral hole-burning materials, and other materials [5].

In photorefractive materials [1], [14]-[17], an interference pattern (or a hologram) which results from the interference of an object beam with a reference beam. Since the object beam carries an image containing a 2-D bit array (a data page), the capacity and data access rate are potentially high. To retrieve the recorded pattern, a reconstruction beam reads out the data image from the interference pattern. The large capacity of photorefractive materials is achieved by multiplexing a number of holograms in the same volume, and each contains a large 2-D bit array. Multiplexing schemes such as angular [16]-[19], wavelength [19]-[21], phase-code [22], [23], spatial [24], fractal [25], and peristrophic [26] multiplexing have been developing to increase the number of multiplexed holograms with lower cross-talk. In order to explore further the available capacity of the holographic material, more than one multiplexing scheme may be combined, such as spatial-angular [27], fractal-spatial [28], and spatial-angular-peristrophic [29] in a volume.

The two-photon effect is based on the absorption of two optical beams for write-in or read-out of data [30], [31]. The advantage of two-photon technology is because the energy required to change the states is small (< 10 fJ/μm^2), the data access time is also relatively small (≈ 30 ps).

In spectral hole-burning materials [32], [33], e.g., chlorin, the absorption coefficient and/or refractive index are spatially modulated by absorption of the incident light at a particular absorption band of the molecule. Because this type of materials contains inhomogeneous molecules which have different absorption bands, and each reacts to a different wavelength, high storage capacity can be achieved using multiple wavelengths recording.

Output Interface

The parallel output of the memory medium impinges on a photodetector array or CCD camera at the left side, and is converted to electrical form. In current OPOM systems, a CCD device consisting of one million pixels has been used [34]. However, the data is transferred at TV frame rates which are far below the rate of a high speed network. To achieve high aggregate data rate, an array of smaller CCD devices may be used to provide parallel outputs. The error correction systems next to the CCD or photodetector array decodes the data and corrects errors. At the right side, a transmitter array transfers the corrected data to an output channel or network. In the following sections, the error correcting interface will be discussed in detail.

Source of Noise, and Its Effect on OPOM's

The noise sources of the OPOM systems are divided into system and material noises [35]. The system noises include: input/output (I/O) device imperfection, detector noise (thermal and shot noise), lens aberrations, scattering and multiple reflections from lenses and other optical components, misalignment from transmitter to photodetector pixel, and laser non-uniformity and fluctuations. By carefully designing the components and precisely aligning the system, the bit-error rate (BER) is estimated in the range of 10^{-4} to 10^{-5}, which mainly depends on the number of pixels in the I/O devices [34], [35]. Note that even without any data recorded by the memory medium, the raw BER of an OPOM system for using large data pages could not satisfy the low BER requirement of 10^{-15} or better required for digital applications. The BER degrades rapidly to the range of 10^{-6} to 10^{-3} when the medium starts recording holograms multiplexed by the schemes

described above [1], [34], [35]. Such a high BER necessitates the use of error correcting codes in the OPOM for most applications. Examples of the material noise sources are: cross-talk between recorded holograms, inter-pixel cross-talk in a hologram, internal reflections in the medium, non-uniform diffraction efficiency, distortions due to surface imperfection, blurring due to limited spatial resolution, damage to the medium, and scattering by defects and particles. In Section 3, improvement in the BER from the use of error correction coding/decoding are shown.

2.3.2 Optoelectronic Smart Pixel Technology

The advantages of using SP technology are three-fold: parallel operations provide a large data processing rate; optical I/O and free-space interconnections reduce the communication hardware and increase the number of available I/O ports; and electrical circuitry can perform relatively complicated computations and logic. Various SP techniques have been investigated, and each has characteristic strengths and weaknesses. These SP devices can be divided into passive spatial light modulators (SLMs) [36]-[40], active optical sources [41], [42], and hybrid OEIC devices [43], [44].

One type of passive SP devices are SEEDs (self-electro-optic-effect devices) using multiple quantum well (MQW) technology [36]. The basic element of SEEDs is an electrically biased optically controlled PIN diode combining photodetector, switch and modulator. In order to increase the switching speed, later SEEDs have been fabricated with field-effect transistors (FETs) in the same substrate, a combination called FET-SEEDs [37]. In the latest SEEDs, called CMOS-SEEDs, the GaAs-AlGaAs MQW modulators are flip-chip bonded on to wired active silicon CMOS circuits [38]. In this study, we concentrate on the use of CMOS-SEED technology because of its high signal processing capability, low power dissipation, high fabrication density, and well-defined design procedures.

2.4 Error Correction Using Reed-Solomon Codes

Error correction/detection techniques have been widely used in applications such as digital communications, mass data storage, optical compact disk data storage and computers to improve the reliability of information processing [7]. Error correction techniques add or encode a small portion of redundant information into the digital messages before they are transmitted or stored. At the receiver, the received data is decoded, and certain types of errors occurring due to system noise and material defects can be corrected by recombining the error-correcting information. In this study, the Reed-Solomon (RS) code, one class of error-correction codes, is used because of its ability to correct both burst and random errors and because of its great flexibility in code length and properties. Various RS codes can have different numbers of information symbols (their length) while retaining the same error-correcting capability, while many other error-correcting codes are restricted to their length. This section briefly describes the fundamentals and applications of Reed-Solomon codes. Details are contained in Refs. [45]-[47].

2.4.1 Reed-Solomon Codes

Reed-Solomon (RS) codes are one type of block error-correction codes. In block codes, every k *message symbols* are grouped into a block, and $n - k$ redundant *parity-check symbols* are appended to the message symbols. This forms an n-symbol *data block* and each block is called a *codeword.* Because the parity-check symbols are linear combinations of the message symbols within the same block, each block is generally uncorrelated with the others. RS codes are defined in finite fields, or called Galois fields. A finite field is denoted by $GF(2^m)$ when there are 2^m distinct elements in the field.

An (n, k) Reed-Solomon (RS) code from $GF(2^m)$ that corrects at most t symbol errors has the following parameters:

Codeword length: $n = 2^m - 1$,
Number of parity-check symbols: $n - k = 2t$,

Minimum distance: $d_{min} = 2t + 1$.

Each GF(2^m) element or code symbol in RS codes consists of m binary elements, and thus each RS codeword consists of mn binary bits including $2mt$ parity-check bits. The *code rate* r is defined as the ratio of the number of *information* (*message*) *symbols* to codeword length n, i.e.,

$$r = \frac{k}{n} = 1 - \frac{2t}{n}. \qquad (1)$$

2.4.2 Coding of Reed-Solomon Codes

In a bit stream of binary data, every km bits are grouped into a k-symbol message block and each symbol contains m bits. Let $(u_0, u_1, \cdots, u_{k-1})$ be a message block where $u_i \in$ GF(2^m). This block of symbols is also expressed in polynomial from as $u(x) = u_0 + u_1 x + \cdots + u_{k-1}x^{k-1}$. One way to construct (encode) the t-error-correcting RS code is to multiply a message polynomial by a *generator polynomial*

$$g(x) = \prod_{j=1}^{2t} (x - \alpha^j) \qquad (2)$$

with roots of $2t$ consecutive powers of α where α is a primitive element in GF(2^m). That is

$$v(x) = u(x) \cdot g(x) = v_0 + v_1 x + \cdots + v_{n-1}x^{n-1}.$$

However, the RS codes usually appear in a systematic form rather than the above. A systematic RS codeword contains two parts, k information symbols and $2t$ parity-check symbols. The $2t$ parity-check symbols are the coefficients of the remainder which results from dividing the information polynomial $x^{2t}u(x)$ by the generator polynomial $g(x)$.

When decoding a retrieved RS code from a communication channel or a data storage, a *syndrome* containing $2t$ symbols is calculated from the retrieved codeword. Each syndrome corresponds to an error pattern which corrupts the original codeword during transmission. The error pattern results from an error-locator polynomial which is obtained from the syndrome using a modified Euclid's algorithm. The location of errors is the reciprocal of the roots of the error-locator polynomial. The decoding scheme adopted in this study is the transform decoding algorithm (TDA) [12]. Another decoding scheme is to record all the error patterns in a look-up table which is accessed by the calculated syndrome. However, the look-up table scheme is useful only for RS codes with short length because of the extensive hardware involved in building a huge table for long RS codes.

2.4.3 Performance of the Reed-Solomon Codes

The performance of the Reed-Solomon codes is evaluated by the output bit-error probability, or bit-error rate (BER). An upper bound of the output BER for a t-error-correcting (n, k) RS code in GF(2^m) is given by [48]

$$P_e \le \frac{2^{m-1}}{2^m - 1} \sum_{j=t+1}^{n} \frac{j+t}{n} \binom{n}{j} \cdot P_s^j (1 - P_s)^{n-j}, \qquad (3)$$

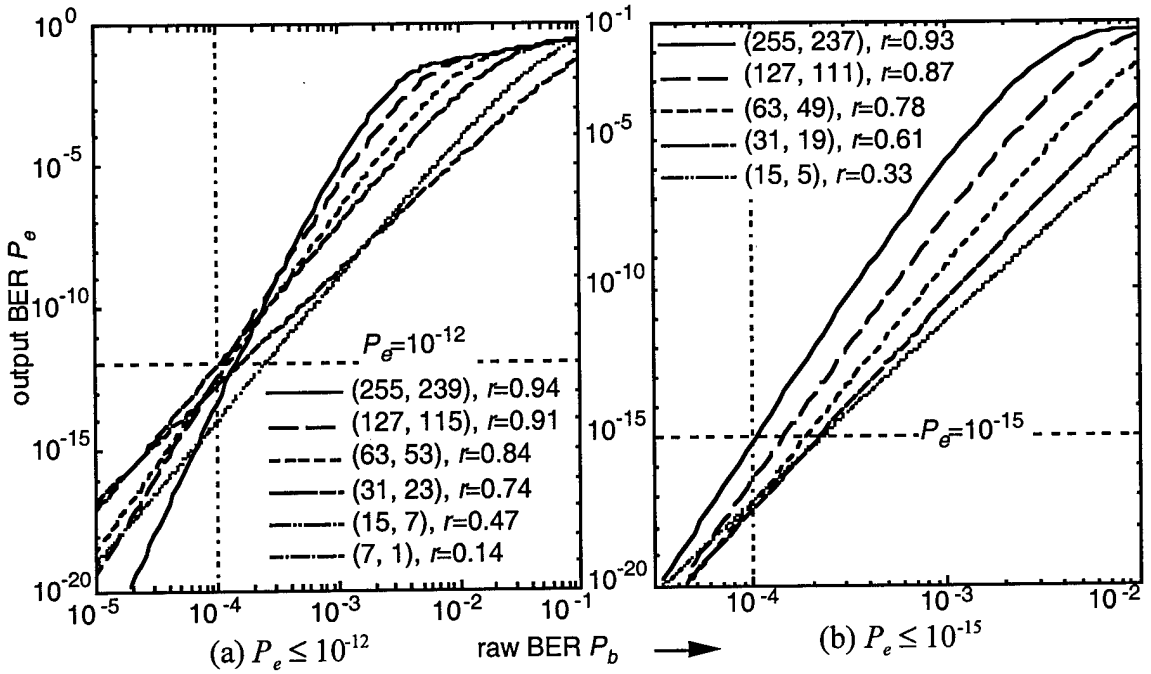

Figure 2.3 Performance of primitive RS codes for (a) $P_e \leq 10^{-12}$ and (b) $\leq 10^{-15}$ at $P_b = 10^{-4}$.

where P_b is the raw (input) bit-error probability and $P_s = 1 - (1 - P_b)^m$ is the symbol-error probability. *Primitive* RS codes are defined as those in which n is exactly equal to $2^m - 1$. In practice RS codes of any length n' (so-called *truncated codes*) can be constructed using a selected subset of information symbols within a primitive code. Figures 2.3 (a) and (b) show the performance of primitive RS codes which reduce the output BER below 10^{-12} and 10^{-15} respectively with a raw BER of 10^{-4}. The parameters of these RS codes are listed in Table 1 showing that, for

$10^{-4} \rightarrow 10^{-12}$				$10^{-4} \rightarrow 10^{-15}$			
(n, k)	m	t	r	(n, k)	m	t	r
(7, 1)	3	3	0.14	(non existent)	3	-	-
(15, 7)	4	3	0.47	(15, 5)	4	5	0.33
(31, 23)	5	4	0.74	(31, 19)	5	6	0.61
(63, 53)	6	5	0.84	(63, 49)	6	7	0.78
(127, 115)	7	6	0.91	(127, 111)	7	8	0.87
(255, 239)	8	8	0.94	(255, 237)	8	9	0.93

Table 2.1 Parameters for the Reed-Solomon codes which reduce BER from 10^{-4} to 10^{-12} and 10^{-15}.

codes of similar capability, the code rate of the RS codes with longer codewords is always larger. Thus RS codes with short codeword length are less efficient in the use of information space and bandwidth. Figure 2.4 shows the code rate r vs. codeword length n for different primitive and truncated RS codes. The primitive codes exist for $n = 3, 7, 15, 31, 63, \ldots$ etc., and are plotted as discrete points for values of required output BER (P_e) of 10^{-9}, 10^{-12} and 10^{-15} with a raw BER of 10^{-4}. These points are connected by lightly dotted lines to show the general trend, even though primitive codes do not exist on these lines. The other curves show the results for truncated RS codes for $m = 5$ (Fig. 2.4(a)) and $m = 8$ (Fig. 2.4(b)).

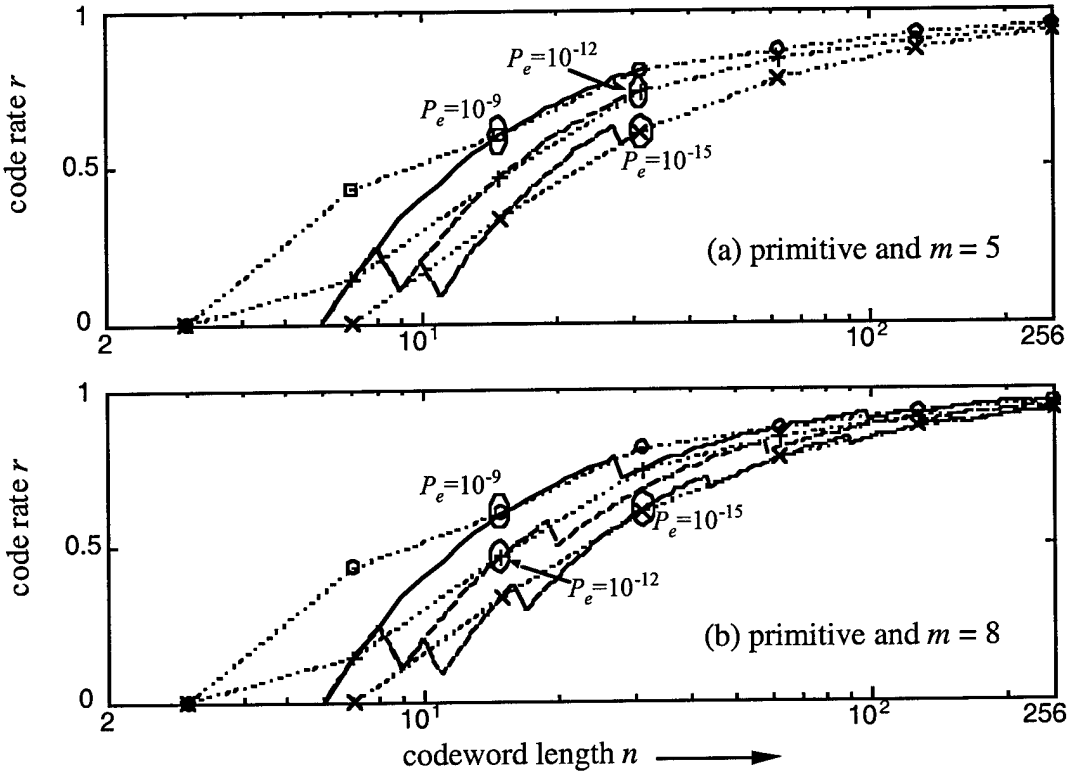

Figure 2.4 Code rate versus codeword length for primitive RS codes (dotted lines), (a) $m = 5$, and (b) $m = 8$ RS codes at a raw BER $P_b = 10^{-4}$.

2.4.4 Applications of the Reed-Solomon Codes

When Reed-Solomon codes are applied to digital systems, simple, regular RS codes are seldom used. In practical applications, interleaving and combining of the RS codes is frequently used. The interleaving of several codewords breaks up a burst error into several shorter ones and, thus, makes correction of the burst easier. The combining of two codes improves the error-correction capability and makes the decoder design simpler. (See [7], [8], [45], [49] for applications in detail.)

Burst errors resulting from material defects or particle noise may corrupt hundreds to thousands of data bits during transmission or recording. To break a huge burst error into smaller ones, a group of codewords is transmitted in such a way that the code symbols of the same order are sent into the channel in sequence, than the next order symbols, and so on. The number of codewords in the group is called the interleaving depth, and its selection depends on the size of the expected burst error.

Combining two smaller codes can simplify the decoder design and improve the error correcting capability. In turn, the code efficiency or the code rate is increased while maintaining the same error correcting capability. Concatenated, cross-interleaved, and product codes belong to this category. When encoding, for example, the data of a product code are arranged in a $k_1 \times k_2$ array. The k_2 columns each of k_1 elements are first encoded individually by an (n_1, k_1) encoder which results in an $n_1 \times k_2$ array. Then, the rows are encoded by an (n_2, k_2) encoder resulting in an $n_1 \times n_2$ code array. The decoding is performed in reverse order by decoding the rows of the code array followed by the columns.

2.4.5 Parallel Reed-Solomon Decoder: The Previous Work

Error correction coding techniques have been used to reduce the bit-error rate at the output of optical page-oriented memories. In addition, the use of error correction can increase the usable capacity of the memory [6]. The output signal-to-noise (SNR) ratio and the bit-error rate (BER) degrade proportional to the number of data pages recorded in holographic memory media. At a desired BER, only a certain number of data pages can be recorded. However, using a small portion of the memory capacity for the error-correction codes, more data pages can be recorded since the degraded output BER is decreased by decoding the readout data. As an example, the output SNR was about 20 for 40 holograms recorded in a Cu-doped KNSBN crystal. The SNR degraded to 3 when 100 holograms were recorded. When a Reed-Solomon code of code rate 0.8 was used to encode the input data, the decoded SNR of the 100 holograms improved to 20. The effective memory capacity, equal to 80 (= 100 × 0.8) holograms, was twice that of the uncoded memory.

A parallel electronic RS decoder with 60 optical inputs for a (15, 9) RS code in $GF(2^4)$ was proposed, and it provided an effective data throughput of 300 megabits per second [9], [10]. In a fixed VLSI area of 10 cm^2, with an output BER below 10^{-12} at a raw BER of 10^{-4}, the OE parallel RS decoder for the $GF(2^6)$ RS code (primitive) of codeword length 63 provided the largest effective data throughput. The parallel RS decoders were able to provide a data throughput up to 10^{12} bits per second if a 0.1-μm CMOS process were used in the fixed area. Compared with an array of conventional symbol-serial RS decoders (called bit-parallel/symbol-serial (BPSS) decoders here), the OE parallel RS decoder provides larger data throughput and more efficient utilization of VLSI area.

2.5 Smart Pixel Interfaces for Optical Page-Oriented Memories

Optical page-oriented memories (OPOMs) require input/output (I/O) interfaces to transfer large amounts of information without resulting in an I/O bottleneck. This section discusses the application of smart pixel techniques to construct interfaces between OPOM systems and data transmission networks having satisfactory aggregate data rates. The following sections describe a performance analysis and outline system characteristics, assumptions and symbols used in the design of an OPOM system.

2.5.1 Introduction

A schematic diagram of an OPOM and its components was given in Section 2.A. Here, we summarize additional functions of the interface components:

i) Format conversion. The conversions include serial-to-parallel/parallel-to-serial (spatial) and wavelength conversions. Because the number of I/O channels in OPOMs differs from that in physical network and interconnection hardware by several orders of magnitude, the OPOMs need spatially demultiplexing and multiplexing at the input and output interfaces, respectively. Wavelength conversion is needed because of different optical wavelengths used in OPOM storage materials and optoelectronic SP devices.

ii) Interface to wave guides and free-space input/outputs. Direct coupling of optical signals from/to array of waveguides (e.g., optical fibers) requires precise alignment, and relay lenses and lenslet arrays are required to transfer optical signals.

iii) Error encoding/decoding. Due to the inherent noise and crosstalk in memory materials and OE components, errors will occur at the memory output. The SP interface needs error control coding techniques to correct errors, increase the reliability of the retrieved data, and decrease output BER.

Because the decoding processes are more complicated than the encoding process, we concentrate on designs of the output interfaces. Figure 2.5 shows the conceptual structure of an output interface performing the photo detection, error correction, and parallel-to-serial conversion. A CCD array or an array of photodetectors receives an optical data page retrieved from the memory medium. Error-correction decoders implemented by electrical circuitry are connected to the photo detector array and decode the retrieved data. The decoding of retrieved data requires the most hardware and results in the longest delays, as we describe in the following sections. An array of many-to-one spatial multiplexers performs the parallel-to-serial conversion, and an optical transmitter array converts the electrical signal into optics. The multiplexing can be achieved using shift registers. An array of shift registers are grouped, and each group is associated with an optical transmitter. The shift registers first buffer the decoded data bits. Then the transmitter converts the decoded data bits into an optical signal and transfers them to output port in sequence.

The array of error-control decoders in the output interface is required to provide high data throughput and low output bit-error probability. We assume that each data page contains 1,024 × 1,024 bits and is accessed in 10 μs. The data throughput is then 10^{11} bits per second, i.e., 0.1 terabit per second. We choose these projected limits as representative of those for OPOM systems in the next few years. The uncoded BER is assumed to be 10^{-4}, and the output BER is required to be 10^{-9} to 10^{-15}. Our goals are to develop design schemes for the SP interfaces that satisfy the two requirements simultaneously under limits of fixed chip area (assumed 10 cm^2) and fixed power dissipation (assumed 1 to 5 Watts per cm^2), and to define parameters to evaluate the performance.

In Fig. 2.5, the serial Reed-Solomon decoders, for example, are assumed to decode retrieved data bits in this structure. The input data page contains 1,024 × 1,024 bits. The data page is divided into 32 × 32 blocks. Each block is further divided into 4 sub-blocks and so each contains 256 input channels. The number of sub-blocks is determined by the number of data bits which can be decoded by a single decoder in a memory access period, i.e., 10 μs. In each multiple RS decoder unit, there are 4 serial RS decoders, and each is electrically connected to a sub-block of input channels. The outputs of the four RS decoders are multiplexed by a 4-to-1 parallel-to-serial converter, and finally the electrical signal is converted to an optical output by an optical transmitter.

2.5.2 Implementation of Transform-Decoding-Algorithm Reed-Solomon Decoders

We have studied several implementations of the RS decoder using the transform decoding algorithm in order to compare the effect of I/O parallelism. The TDA RS decoder with symbols input sequentially is called the symbol-serial RS decoder [12], and is called the symbol-parallel RS decoder when all the symbols of a codeword are simultaneously input to the decoder. In addition, the multiplier of two finite field symbols (**FFM**), a key component in RS decoding processes, is implemented so the bits of a symbol either sequentially or simultaneously are input to or output from the multiplier [59]. With these alternative I/O formats of bits and symbols, there are four basic RS decoder designs: bit-serial/symbol-serial (BSSS); bit-parallel/symbol-serial (BPSS); bit-serial/symbol-parallel (BSSP); and bit-parallel/symbol-parallel (BPSP). We implemented FFMs by using 1-D and 2-D systolic arrays so the decoding rate is increased, at the expense of increased hardware complexity. By trading off the decoding rate and hardware complexity, a multiplier with

bits simultaneously input is implemented using compound logic circuits. As a consequence, six RS decoder designs are studied, and they are denoted by BSSS, BPSS-S, BPSS-C, BSSP, BPSP-S, and BPSP-C. Here, the S and C suffixes represent systolic array and compound-gate FFM, respectively.

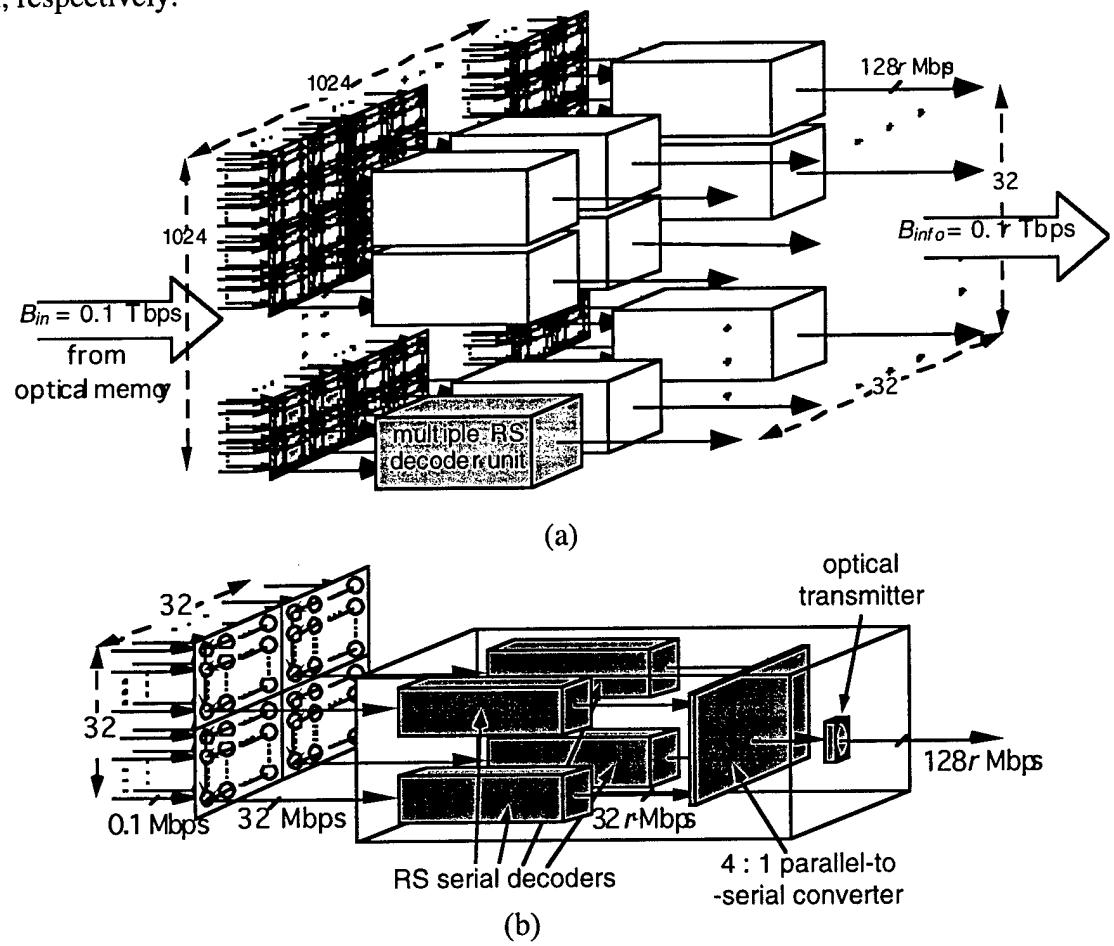

Figure 2.5 (a) Conceptual structure of the smart pixel interface using serial decoders, and (b) a multiple decoder unit. We assume 1024 ¥ 1024 inputs (0.1 Mbps/channel) and 32 ¥ 32 outputs (128*r* Mbps/channel). The input page is divided into 32 ¥ 32 blocks, and each contains 32 ¥ 32 bits. Each block requires 4 serial RS decoders, for example, to decode the 1024 bits in 10 μs.

In Table 2.2, the properties of these designs are summarized. The first three rows of Table 2.2 list the acronym, the I/O formats of an input codeword, and the types of multipliers used in the implementations. The codeword delay in the fourth row is defined as the longest delay needed by the slowest operation in the implementation for a codeword. The reciprocal of the codeword delay is proportional to data rate of the output interface, a parameter which affects the total data throughput. In the BSSS and BPSS designs, the slowest operation is the inverse transform of the error sequence (**ITES**) which needs $N = 2^m - 1$ symbol delays because of the sequential input of N symbols. Unfortunately, the delay of this module does not decrease for any shortened codes because an error sequence always contains N symbols. Since each type-1 FFM consists of m cells and each cell needs a unit delay, a symbol delay of the BSSS implementation corresponds to m unit delays. In consequence, the BSSS needs mN unit delays to process a RS codeword. On the other hand, the bit-parallel FFMs simultaneously process m bits of a symbol, and thus need only one unit delay for each symbol, i.e., 1 symbol delay = 1 unit delay (BPSS). Therefore, the BPSS implementations which process a code symbol in parallel need N unit delays between two

consecutive codewords. In the symbol-parallel implementations, the slowest modules are the polynomial normalization and/or the ITES modules. Each cell in the latter module needs N parallel FFMs followed by an N-input m-stage adder which results in about 2 symbol delays. The module of polynomial normalization needs 2 symbol delays by assuming one symbol delay to the computation of the reciprocal of a finite field element. Therefore, a codeword delay of the symbol-parallel implementation contains two symbol delays, or corresponding to $2m$ and 2 unit units. Note that if more register stages are placed in the buffers of the symbol-parallel implementations, then only one symbol delay is needed between two codewords.

RS decoder	BSSS	BPSS-S	BPSS-C	BSSP	BPSP-S	BPSP-C
I/O format	bit-serial symbol-serial	bit-parallel symbol-serial	bit-parallel symbol-serial	bit-serial symbol-parallel	bit-parallel symbol-parallel	bit-parallel symbol-parallel
FFM type	type-1	type-s2/s3	type-c2/c3	type-1	type-s2/s3	type-c2/c3
codeword delay	mN	N	N	$2m$	2	2
delay from 1st input to 1st output	$2m(2N+n+2)$	$2m(2N+n+2)$	$2(2N+n+2)$	$2m(N+4)+N-2t+2$	$2m(N+4)+N-2t+2$	$3N+10$
numbers of I/O	1/1	m/m	m/m	n/k	mn/mk	mn/mk

Table 2 .2 Properties of the implementations of the RS decoder using the transform decoding algorithm

The delay between the first input symbol of a received codeword and the first output of its correction is listed in the fifth row. This delay is the part of memory access time that is consumed by the output interface. Note that these formulas are simplified by assuming $\log_2 N \approx m$ and $\log_2 t \approx 1$. Also, note that this delay does not vary as a function of FFMs used because the total delay for the three systolic FFMs are the same, are the compound-circuit FFMs.

The number of transistors per decoder for two families of RS codes, $m = 5$ and 8, is shown in Fig. 2.6. The selected shortened and extension codes here have the same performance as the primitive codes which are discussed previously and shown by the dotted lines. As n decreases in the shortened codes, the number of transistors per decoder only reduces slightly because the number of transistors of the ITES mainly depends on the natural length $N = 2^m - 1$ and the ITES module needs the most transistors among these decoding modules. In addition, the number of transistors of other modules depends on t and m more than on n. Therefore, we conclude that the TDA scheme is not suitable for the implementation of shortened RS codes in all the decoder designs studied.

2.6 System Analysis of Smart Pixel Interfaces

In this section, novel parameters are defined and the performance of the decoders and the feasibility of the SP interfaces is analyzed using computer simulations. Given a set of RS code parameters (m, n, k, t, r), the number of logical gates and transistors of a TDA RS decoder are calculated from Table 2. Then, the chip area, power dissipation, and maximum clock frequency of the VLSI decoding implementations are estimated using the modified SUSPENS model discussed in Appendix A. The parameters used in the modified SUSPENS are listed in Table 3, which is obtained from the current VLSI CMOS processes with proper modifications. For example, the Rent's constant p for high speed microprocessor chips is typically from 0.6 to 0.7, depending on the interconnection complexity of the circuits. Note that p is empirical. Therefore, we assumed p

= 0.6 for the BSSS implementation because of serially connected modules and cells, and p = 0.68 for the BPSP-C because of the more global interconnections in the parallel decoding modules and compound-circuit FFMs. The p's of other implementations are then specified with values between these two extremes. In particular, the p of the buffers is 0.5 because D-flip-flops are sequentially connected to each other and only local connections are used. The values of the parameters marked with an asterisk are obtained by assuming 0.8-μm CMOS, and they are scaled when different CMOS feature sizes are applied.

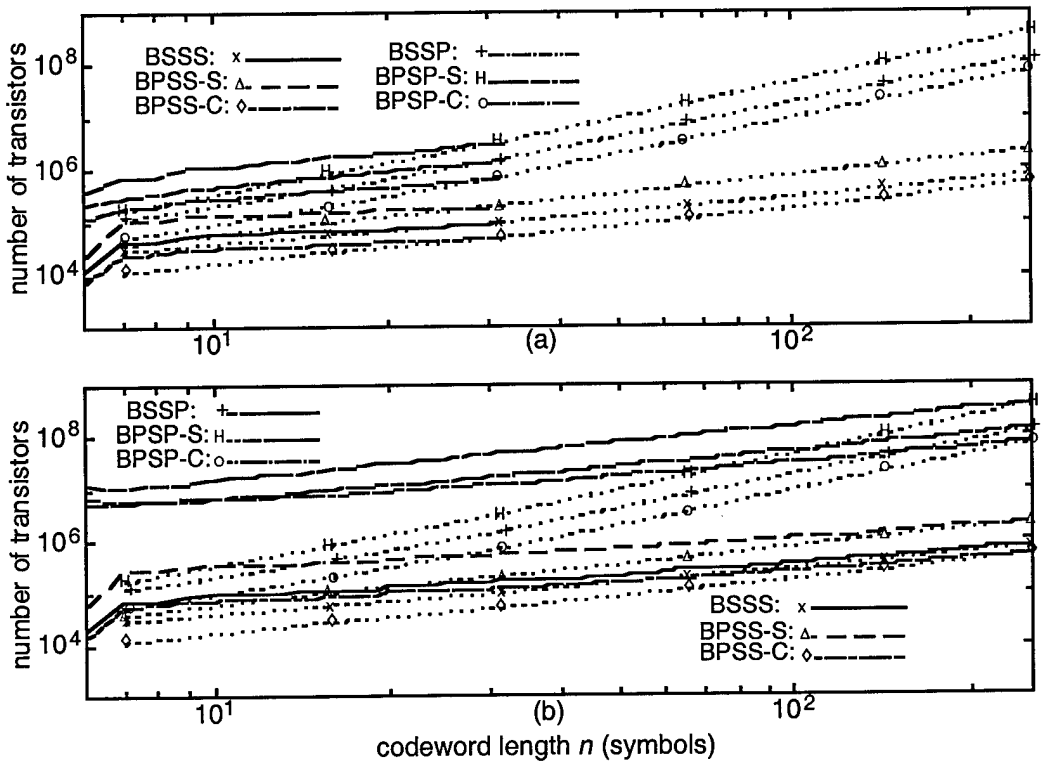

Figure 2.6 The number of CMOS transistors per RS decoder versus codeword length for primitive RS codes which reduce the BER from 10^{-4} to 10^{-12} (dotted lines), (a) m = 5 RS codes, and (b) m = 8 RS codes.

2.6.1 Performance of An Individual TDA RS Decoder

Using the modified SUSPENS VLSI simulation model and the circuit parameters, the chip (decoder) area and power dissipation of a decoder and modules were estimated. In the following estimates, the parameters assumed include 0.25-μm minimum feature size (F), 10^{-4} raw BER (P_b), and 10^{-12} output BER (P_e), except for those listed in Table 2.3.

In Fig. 2.7, the decoder area needed for the primitive RS codes ($n = 2^m$-1) are shown by dotted lines and symbols at the discrete positions where they exist, while the other lines show the area of the TDA decoders for the shortened and extended RS codes of m = 5 and 8. Similar to Fig. 2.6, the decoder area needed for the TDA decoders separate into two groups, the symbol-serial (BSSS, BPSS-S, and BPSS-C) and the symbol-parallel (BSSP, BPSP-S, and BPSP-C), as n increases. For the primitive codes of small n, the area of the symbol-parallel decoders is an order of magnitude larger than that of the symbol-serial decoders, and it is three orders larger for large n. However, the area changes slightly for the RS codes with the same m.

parameters	buffers	BSSS	BPSS-S	BPSS-C	BSSP	BPSP-S	BPSP-C
p_w (μm)	$3F$	$3F$	$3F$	$3F$	$3F$	$3F$	$3F$
e_w	0.4	0.4	0.4	0.4	0.4	0.4	0.4
n_w	2	3	3	3	3	3	3
p	0.5	0.6	0.63	0.64	0.64	0.67	0.68
f_g	2	2	2	3	3	4	4
f_{ld}	2	4	4	4	m	m	m
T_g (ns)	2	2	2	2	2	2	2
R_{int} (Ω/μm)*	9×10^{-4}	9×10^{-4}	9×10^{-4}	9×10^{-4}	9×10^{-4}	9×10^{-4}	9×10^{-4}
C_{int} (fF/μm)*	0.2	0.2	0.2	0.2	0.2	0.2	0.2
k_{tr}	3	4	4	4	4	4	4
C_{ox} (fF/μm^2)*	2.3	2.3	2.3	2.3	2.3	2.3	2.3
V_{DD} (volts)*	5	5	5	5	5	5	5
f_d	0.5	0.2	0.2	0.2	0.2	0.2	0.2

Table 2.3 Parameters for the modified SUSPENS model (* for 0.8-μm CMOS).

The power dissipated by these implementations was estimated by using Eq. (A.13) and is shown in Fig. 2.8. These lines distribute similar to the lines of decoder area in Fig. 2.7. We notice that the power dissipation of the three symbol-parallel decoders and the primitive RS codes of long n's, (e.g., 255), which normally yield high code rates, are intolerably large.

2.6.2 The Optimal Implementation of TDA RS Decoder

In this section, parameters are defined and used to evaluate the performance of the implementations of the SP interface. The same VLSI parameters are used as in the previous section. In addition, the implementation is confined to a fixed area A_{pg} (10 cm^2) and a fixed power density P_{pg} (2 Watts/cm^2), and all the codes shown here reduce a raw BER from 10^{-4} to 10^{-12} or lower.

The first parameter of the TDA decoders is the input spatial channel density, d_{scin}, which is defined as the number of input channels per unit area as

$$d_{scin} = \frac{N_{D-A} \cdot [\text{no. of inputs per decoder}]}{A_{pg}}, \tag{4}$$

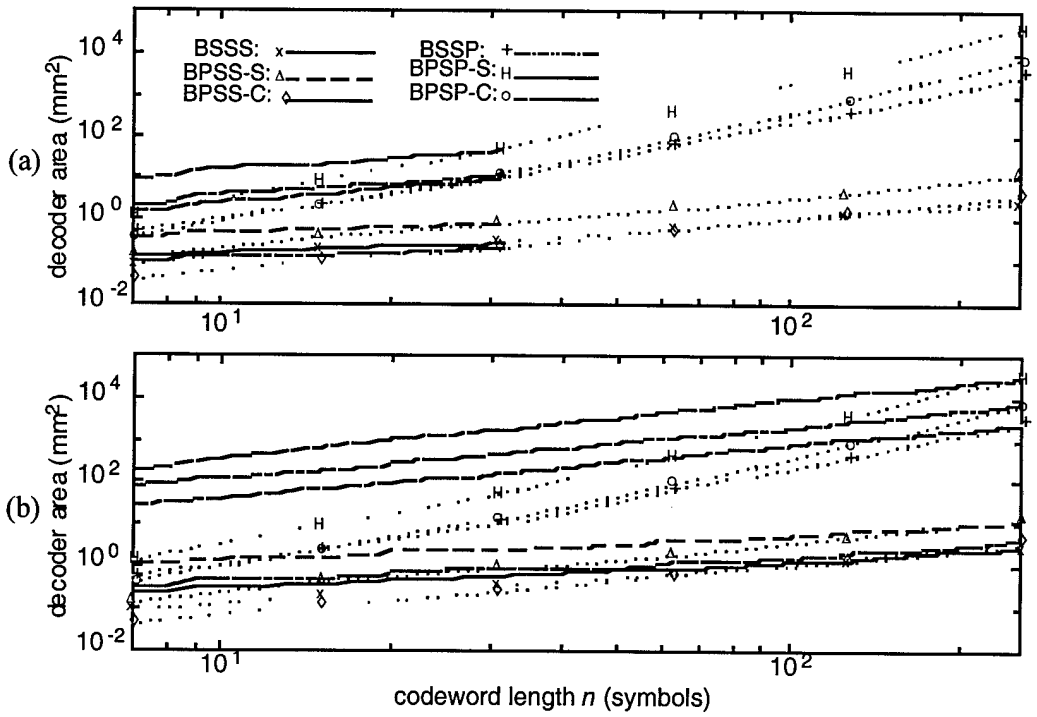

Figure 2.7 Decoder area (mm^2) for primitive RS codes (dotted lines), (a) $m = 5$ RS codes, and (b) $m = 8$ RS codes ($F = 0.25$ μm, $P_b = 10^{-4}$, $P_e = 10^{-12}$). (BSSS: Bit-Serial-Symbol-Serial; BSSP: Bit-Serial-Symbol-Parallel; BPSS-S: Bit-Parallel-Symbol-Serial (using the 2-D systolic FFM's); BPSP-S: Bit-Parallel-Symbol-Parallel (using the 2-D systolic FFM's); BPSS-C: Bit-Parallel-Symbol-Serial (using the compound-circuit FFM's); and BPSP-C: Bit-Parallel-Symbol-Parallel (using the compound-circuit FFM's))

where $N_{D\text{-}A}$ is the number of decoders which are fabricated in A_{pg}. The numerator presents a number of bits N_{blk} which are simultaneously input to the SP interface, and called a **data block**. Figure 2.9 shows the d_{scin} for the primitive, $m = 5$, and $m = 8$ RS codes. Note that the $m = 2$ RS codes do not provide error correction capability of reduction the BER from 10^{-4} to 10^{-12} and thus are not shown. The RS codes with large n result in small d_{scin} due to the logarithmic increase of D_c. In Fig. 2.9 (b), the lines of the BSSP and the BPSP-C stop at $n = 128$ and the BPSP-S at 64 is because the area of a single decoder increases larger than A_{pg} (Fig. 2.7). Therefore, no BSSP, BPSP-C, or BPSP-S decoders can be implemented for those n's. The two horizontal dashed lines in Fig. 2.9 show 1-D and 2-D electrical limits given by the maximum numbers of input pins on the edge of the chip and through the chip, respectively. The 1-D electrical input is limited to is 20 channels per cm^2 (or 40 I/O channels per cm^2 in total), and the 2-D electrical limit is 50 input channels per cm^2 (or 100 I/O channels per cm^2).

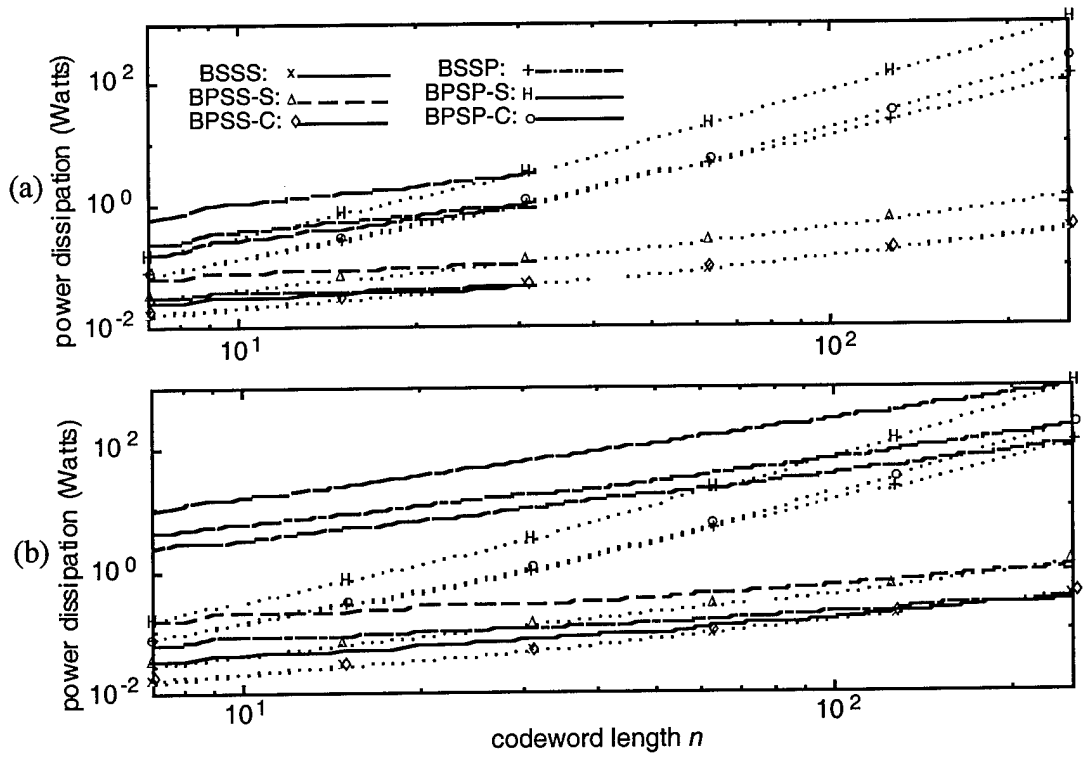

Figure 2.8 Power dissipation per decoder for primitive RS codes (dotted lines) (a) m = 5, and (b) m = 8 RS codes (F = 0.25 μm, $P_b = 10^{-4}$, $P_e = 10^{-12}$).

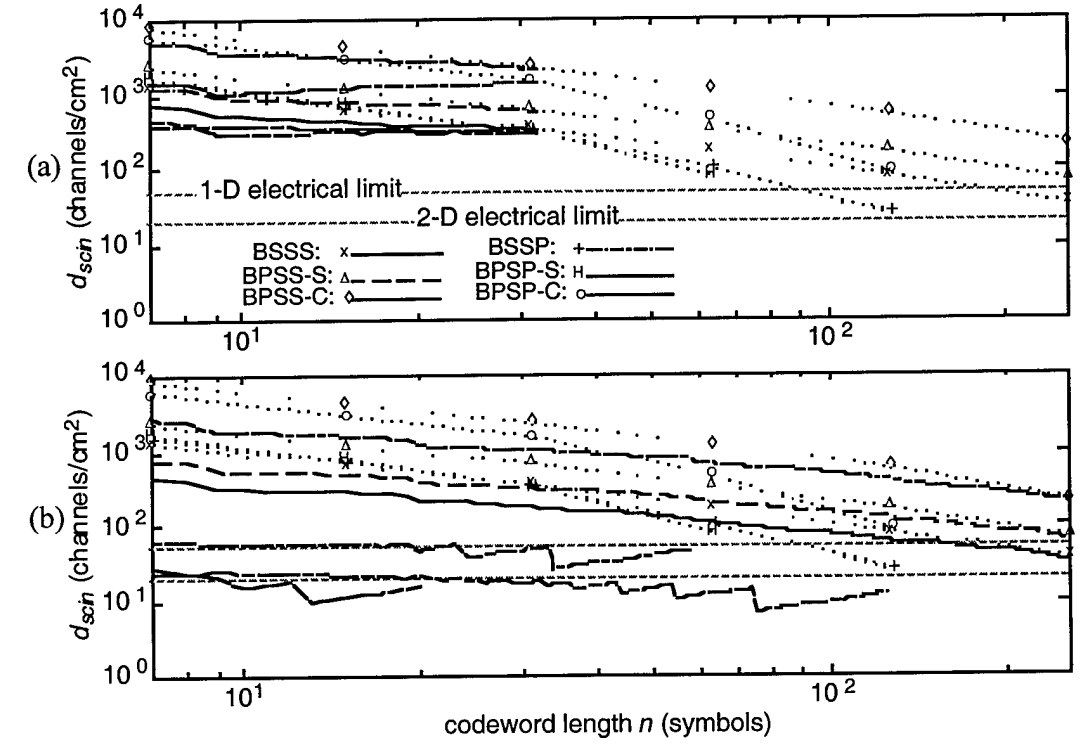

Figure 2.9 Input spatial channel density d_{scin} for primitive RS codes (dotted lines), (a) m = 5 RS codes, and (b) m = 8 RS codes (F=0.25 μm, P_b=10^{-4}, P_e=10^{-12}).

Fixing the power density, the estimated input rate of a data block is given by

$$f_{blk} = \frac{f_c \cdot P_{pg}}{N_{D-A} \cdot P_{RSde}}. \tag{5}$$

Then, the aggregate data rate input to the SP interface, B_{in}, which is the product of the number of bits of a data block and the block rate, f_{blk}, is given by

$$B_{in} = N_{blk} \cdot f_{blk}. \tag{6}$$

Figure 2.10 shows that B_{in} decreases as n and/or m increase. However, the B_{in} of the symbol-parallel implementations decreases slightly because the ITES module, consisting of $(N - 2t)$ cells, dominates the area and the power dissipation and $N \gg t$. Note that because the area of a decoder exceeds the given area A_{pg}, the lines of the BPSP-C, the BPSP-S, and the BSSP of $m = 8$ end at $n = 55$, 18, and 127, respectively. The dashed line at 10^{11} represents a data throughput required by the output interface input from a memory medium.

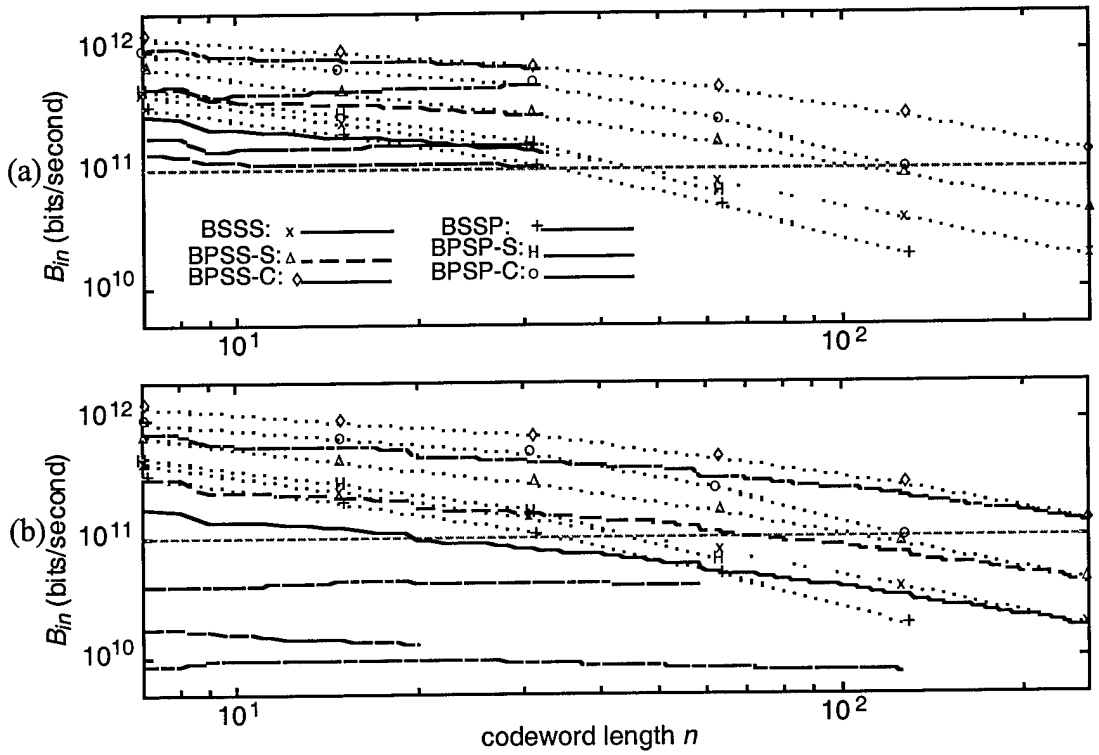

Figure 2.10 Aggregate input data rate B_{in} for primitive RS codes (dotted lines) (a) $m = 5$ RS codes, and (b) $m = 8$ RS codes (F=0.25 μm, P_b=10^{-4}, P_e=10^{-12}).

The information rate at the output of the decoder array (or the aggregate output rate), B_{info}, is given by

$$B_{info} = r \cdot B_{in} = r \cdot N_{blk} \cdot f_{blk}, \tag{7}$$

where r is the RS code rate. The information spatial channel density, d_{info}, defined as

$$d_{info} = \frac{B_{info}}{A_{pg}} = r d_{scin} f_{blk}, \tag{8}$$

is the aggregate output rate in a unit area and is shown in Fig. 2.11. Note that the zigzags in the lines of m = 5 and 8 implementations result from the discontinuities of r, as shown in Fig. 2.4. For the four implementations using the systolic FFMs (BSSS, BPSS-S, BSSP, and BPSP-S), the peak of d_{info} occurs at n = 15 or 31, and the BPSS-S has the largest d_{info} for all n and designs (for the primitive codes). On the other hand, among the four symbol-parallel implementations (BPSS-S, BPSP-S, BPSS-C, and BPSP-C), the BPSS-C and BPSP-C are able to provide better d_{info}. Both peaks of d_{info} of the two TDA decoders occur at n = 31, and, the BPSS-C decoder has the highest d_{info} among the six implementations. The d_{info} of the TDA decoders for m = 5 and 8 RS codes are also shown, and the peak d_{info} are listed in Table 4. In addition, the d_{info} for the codes that reduce the BER to 10^{-15} are shown. In all the cases, the highest d_{info} was obtained by the BPSS-C design at different n, but not at the primitive values, i.e., 2^m-1.

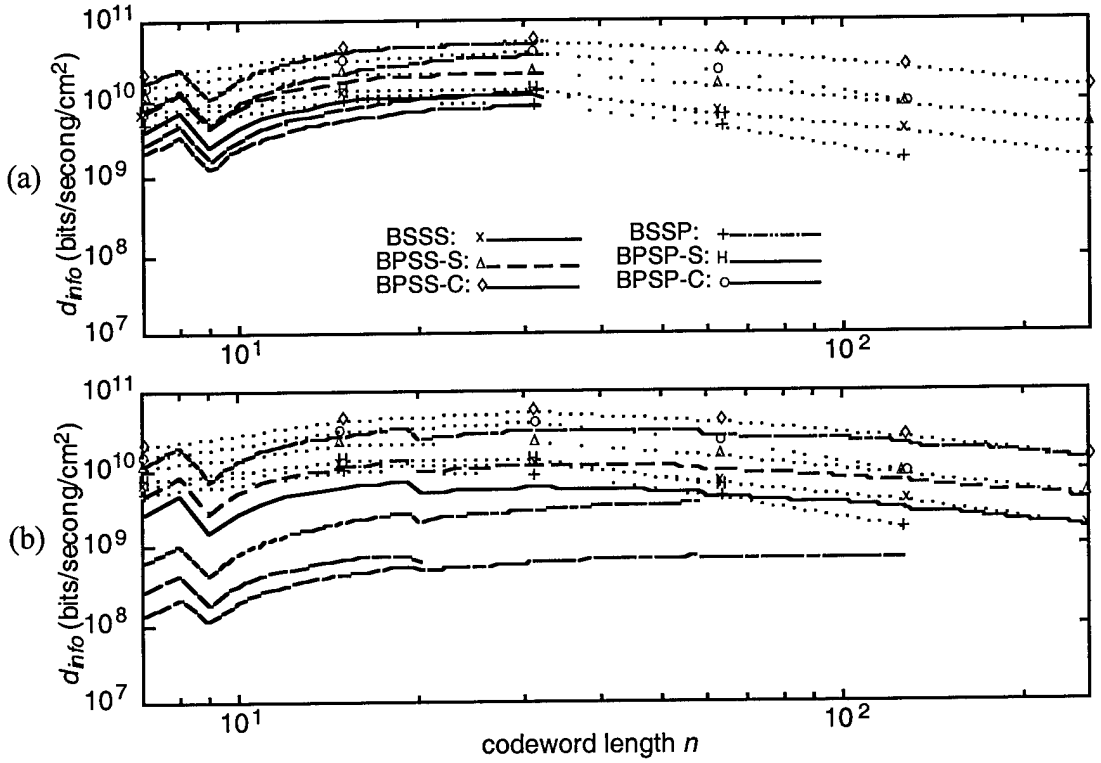

Figure 2.11 Information spatial channel density d_{info} for primitive RS codes (dotted lines), (a) m = 5 RS codes, and (b) m = 8 RS codes (F=0.25 μm, P_b=10^{-4}, P_e=10^{-12}, A_{pg}=10 cm^2, P_{pg}=2 W/cm^2)

2.6.3 Code Dependent Analysis

The RS codes which simultaneously satisfy a specified output BER at a raw BER and a specified code rate are selected by using the code dependent constraints, including BEC and CRC. In order to depict the constraints, an (n, t) code plane is used in which each grid point specifies an RS code.

The bit error constraint (BEC) specifies the minimum number of parity-check symbols in a codeword that is required to achieve the desirable BER. For an RS code, the number of parity-

check symbols is equal to $2t$ where t is the maximum number of error symbols that can be corrected. An upper bound of the output BER P_e of an $(n, n\text{-}2t)$ RS code with raw BER, P_b, is given in Eq. (3). In turn, given m, n, P_b, and P_e, the smallest t satisfying Eq. (3) can be calculated. Figure 2.12 shows the minimum t that is needed to reduce the BER from 10^{-4} to 10^{-9}, 10^{-12}, and 10^{-15} for the primitive, $m = 4$, 5, and 8 RS codes. Note that the t value grows rapidly at small n and becomes steady at large n. This confirms that the long-codeword codes have higher code rate than the short-codeword codes of the same error correction capability.

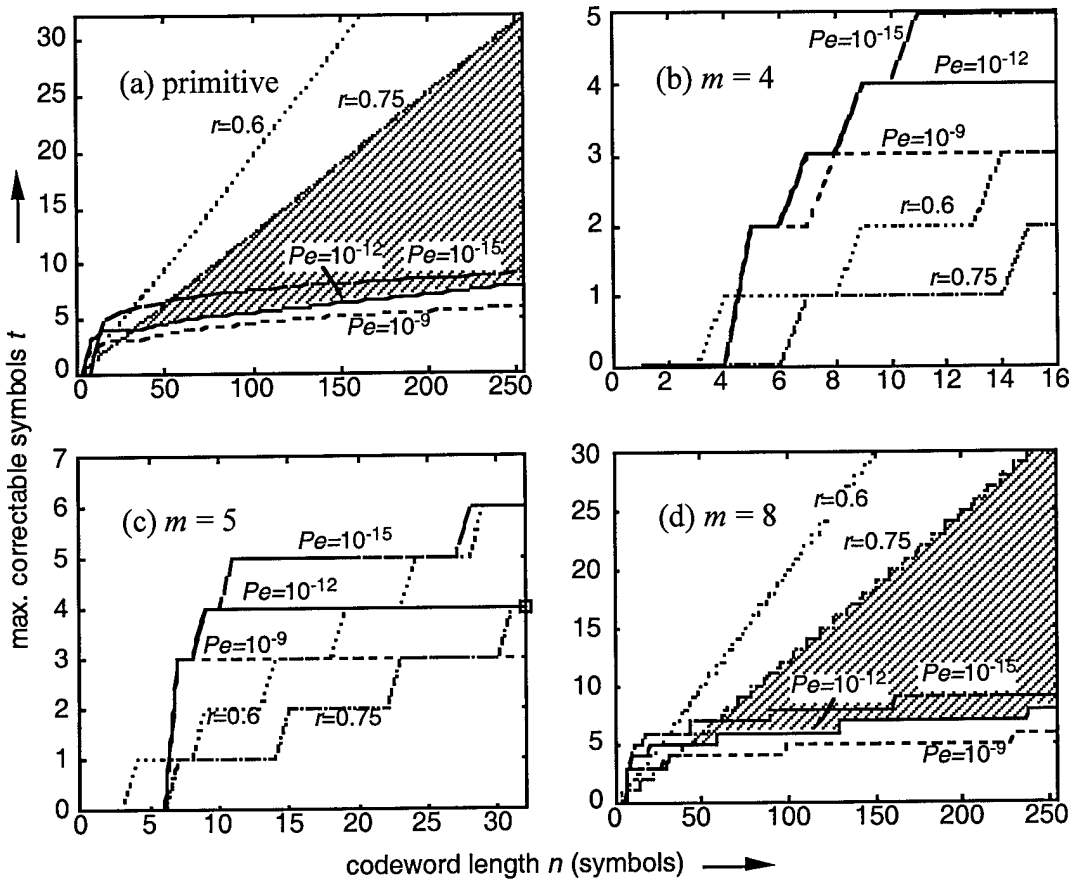

Figure 2.12 The bit error constraint (BEC) and the code rate constraint (CRC) for (a) primitive RS codes, (b) $m = 4$ RS codes, (c) $m = 5$ RS codes, and (d) $m = 8$ RS codes ($P_b = 10^{-4}$). Note that the RS codes in the shaded region simultaneously satisfy $r \geq 0.75$ and $P_e \leq 10^{-12}$.

The code rate constraint (CRC) specifies the maximum t that ensures the code rate r of an $(n, n - 2t)$ RS code is larger than a required code rate r_q. In Fig. 2.12, two dotted lines that specify the maximum t achieving $r = 0.6$ and 0.75 are shown. The RS codes in the intersection of the upper half plane of BEC and the lower half plane of the CRC simultaneously satisfy the two code dependent constraints. For example, the RS codes in the shaded regions in Fig. 2.12 reduce the BER from 10^{-4} to 10^{-12} and have code rate higher than 0.75. Note that no RS codes of $m = 4$ is found, and only the extension RS code in GF(2^5) satisfies the code dependent constraints. In addition, the RS codes on the P_e curve have the highest code rate in the region because of the smallest t (at a fixed n).

	$P_b = 10^{-4} \rightarrow P_e = 10^{-12}$				$P_b = 10^{-4} \rightarrow P_e = 10^{-15}$			
	$m = 5$		$m = 8$		$m = 5$		$m = 8$	
	n	d_{info}	n	d_{info}	n	d_{info}	n	d_{info}
BSSS	26	1.08×10^{10}	19	6.68×10^{9}	27	8.68×10^{9}	38	4.77×10^{9}
BPSS-S	30	2.10×10^{10}	19	1.25×10^{10}	27	1.65×10^{10}	43	9.38×10^{9}
BPSS-C	32	5.16×10^{10}	19	3.26×10^{10}	27	4.11×10^{10}	43	2.63×10^{10}
BSSP	32	8.25×10^{9}	58	7.09×10^{8}	27	6.38×10^{9}	89	6.81×10^{8}
BPSP-S	32	1.22×10^{10}	19	7.83×10^{8}	27	8.97×10^{9}	16	5.24×10^{8}
BPSP-C	32	3.67×10^{10}	32	3.44×10^{9}	27	2.71×10^{10}	43	2.97×10^{9}

Table 2.4 The highest information spatial channel density, d_{info}, of the TDA RS decoders for the $m = 5$ and 8 RS codes.

2.6.4 VLSI Dependent Analysis

In Section 5.B, the block rate was determined by letting all the fabricated decoders operate at the largest possible frequency that is limited by the given power density. The clock rate f_c here, however, was specified to match the data transfer rate at the output channels. The specified clock rate was in general larger than the block rate and, then, only some of the fabricated decoders operated at that rate. Since the input data rate is fixed, choice of the highest code rate becomes the issue, and thanks to the highest code rate determined, the capacity of the optical page-oriented memory is effectively utilized.

There are two VLSI constraints which use the code parameters m, n, and t as inputs to compute the corresponding number of the TDA decoders which satisfy the VLSI physical requirements, the VLSI area A_{pg} and the power density P_{pg}. The first one is called the buffer length or minimum number of decoder constraint (MINC). The minimum number of RS decoders required to provide high data throughput and to prevent access bottlenecks depends on the size of the decoder buffers and the longest decoding delay of a codeword. Given a codeword delay D_{long} (Table 2) and a memory access time t_a, the number of codewords that are processed by an RS decoder in t_a is

$$B_{deW} = \left\lfloor \frac{t_a}{D_{long}} \right\rfloor \text{ codewords,} \tag{9}$$

which corresponds to mnB_{deW} binary bits. Here, $\lfloor x \rfloor$ is the largest integer that is smaller than or equal to x. Note that B_{deW} must be greater than 0, i.e., $t_a \geq D_{long}$, otherwise the selected decoder fails to provide the necessary data rate, and results in an extra access delay. In this case, either other decoder designs are considered or the data-page access time t_a is increased. Because a data page contains N_{in} bits, an interface needs at least

$$N_{D-D} = \left\lceil \frac{N_{in}}{mnB_{deW}} \right\rceil = \left\lceil \frac{N_{in}}{mn\lfloor t_a / D_{long} \rfloor} \right\rceil \tag{10}$$

decoders to process a retrieved data page in a memory access cycle. Here, $\lceil x \rceil$ is the smallest integer that is larger than or equal to x. $N_{D\text{-}D}$ is the minimum number of RS decoders needed by the specified interface.

The second constraint is called the power/area or maximum number of decoder constraint (MAXC). Given an (n, k) RS code over GF(2^m), the decoder area A_{RSde} and the power dissipation P_{RSde} can be estimated using the modified SUSPENS model. Then, in a given area A_{pg}, at most

$$N_{D-A} = \left\lfloor \frac{A_{pg}}{A_{RSde}} \right\rfloor \tag{11}$$

RS decoders can be fabricated. With a limited power density only a certain number of RS decoders can operate at the selected clock rate. Since the power consumed by a decoder is P_{RSde} at a clock rate f_c, the number of decoders that can operate at the same time without excess power dissipation is

$$N_{D-P} = \left\lfloor \frac{P_{pg}}{P_{RSde}} \right\rfloor, \tag{12}$$

where P_{pg} is the power that can be dissipated in area A_{pg}. Therefore, the number of RS decoders that can simultaneously process a data page at a fixed clock rate f_c is given by

$$N_{Dmax} = \min(N_{D\text{-}A}, N_{D\text{-}P}). \tag{13}$$

The $N_{D\text{-}D}$ and N_{Dmax} specified for the primitive RS codes are shown in Fig. 2.13 which is commented as following:

(1) The interface implementation and the RS codes that satisfy all the conditions are determined when $N_{Dmax} \geq N_{D\text{-}D}$. The result shown here agrees with the result shown in Fig. 2.10 in which a dashed line shows the minimum required input rate.

(2) In (a), $N_{D\text{-}D}$ stops at $n = 255$ indicating that the decoding delay D_{long} is larger that the memory access period t_a. Therefore, no minimum number of decoders is specified unless larger t_a is specified.

(3) In figures (b), (d) and (f), N_{Dmax} stops at some n which shows that, beyond that n, either the power dissipation and/or the area of a single decoder are larger than the interface area and the power density, and, therefore, no decoders can be fabricated or operated.

(4) As shown in (d) and (f), the decoding throughput of a single decoder for the codes of large n is larger than the input throughput to the interface. Therefore, only a BPSP decoder is needed. However, the large decoder area and high power dissipation inhibit their fabrication in finite physical conditions.

2.6.5 Interface Feasibility Analysis

To conclude this section , two examples are used to illustrate the design scenario of the smart-pixel error-correcting interface. Both assume: 0.25-μm CMOS process; VLSI area, 10 cm^2; power density, 2 Watts/cm^2; data page size, 1,024 × 1,024 bits; clock rate, 320 MHz (100 MHz for 0.8

μm CMOS process); data page access period, 10 μs; raw BER, 10^{-4}; and output BER, 10^{-12} or better.

The first example shows the result obtained from applying the four constraints to the BPSS-S and BPSP-S implementations for the RS codes in GF(2^5). Figure 2.14 (a) shows the number of TDA BPSS-S decoders in the decoder array in terms of various pairs of parameter (n, t) by applying the MINC and MAXC. Note that the (n, t) can be used for the SP interface when the feasible decoders are more than the required ones, i.e., N_{Dmax} (MAXC) $\geq N_{D\text{-}D}$ (MINC). The intersection of the MINC and the MAXC planes is projected onto an n-t plane, as shown in Fig. 2.14 (b), in which the lines of the maximal t for $r = 0.6$ and 0.75 and the minimal t for $P_e = 10^{-12}$ and 10^{-15} are also shown. It shows that the RS codes of $n \geq 23$ satisfy both $r = 0.6$ and $P_e = 10^{-12}$, and only the (32, 24) code satisfies both $r = 0.75$ and that P_e. Figure 2.14 (c) presents the same result in an n-r plane. Note that no RS codes in GF(2^5) has r greater than 0.75 and reduces the BER from 10^{-4} to 10^{-15}. Figures 2.14 (b) and (c) also shows the intersection of MINC and MAXC of the BPSP-S interface. Although their d_{info} is lower than the BPSS-S decoders (Section 5.B), the intersection is still higher than the r and P_e lines and the BPSP-S decoders can also be used in the SP interface.

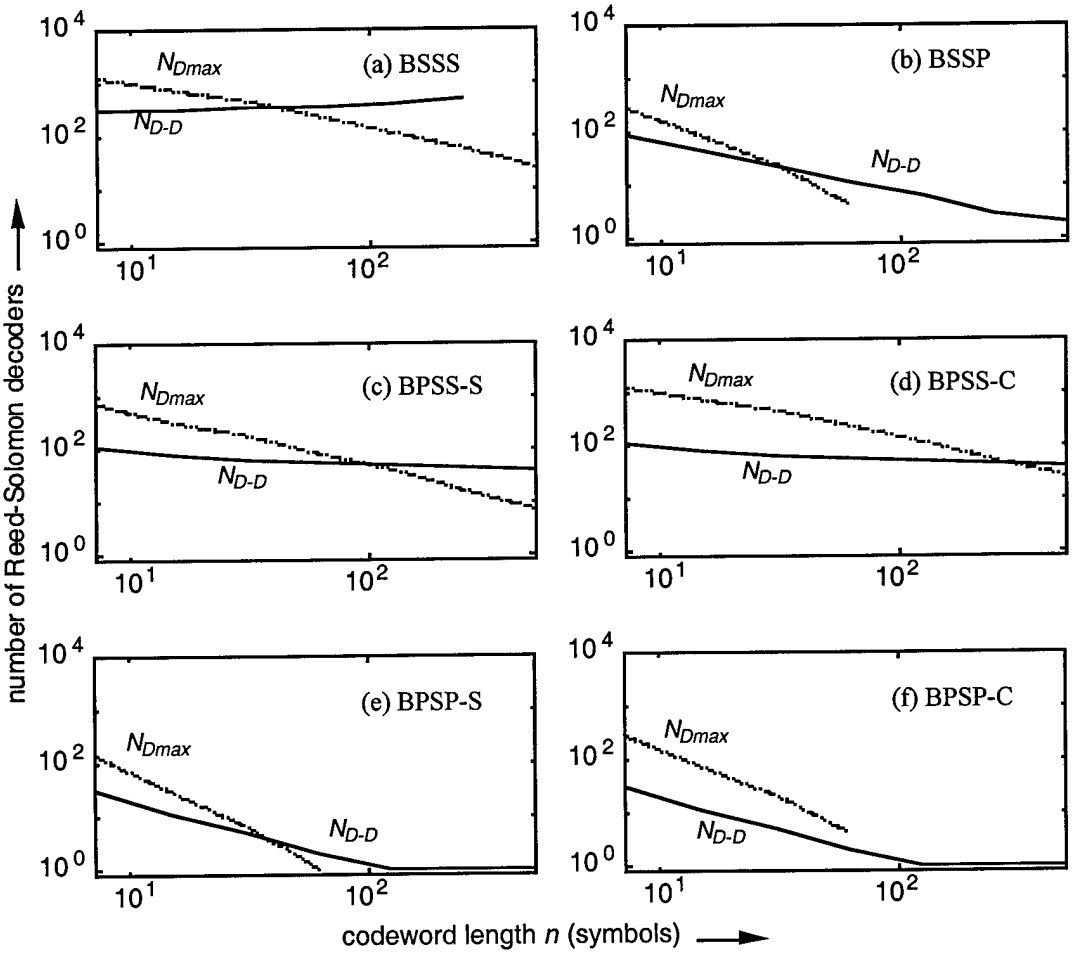

Figure 2.13 The $N_{D\text{-}D}$ from the buffer length constrint (MINC) and the N_{Dmax} from the power/area constraint (MAXC) for the implementations of the (a) BSSS, (b) BSSP, (c) BPSS-S, (d) BPSP-S, (e) BPSS-C, and (f) BPSP-C for primitive RS codes

(F=0.25 μm, A_{pg}=10 cm2, P_{pg}=2 W/cm2, N_{in}=106, P_b=10-4, P_e=10-12, n=2m-1)

In Fig. 2.14 (c), the RS codes in the upper half plane of the intersection curve satisfy the MINC and the MAXC, and the lower half plane of the P_e curve satisfies the BEC. Note that the BPSS-S starts at the top (actually it is from $r = 1$, but not shown here) and the BPSP-S starts at $r = 0$. At small n, the BPSS-S decoder has limited input channels and, hence, the interface requires much more decoders than the area and power can offer. Therefore, limited to the timing and buffer length, no RS codes can be used. On the contrary, the BPSP-S decoder has shorter decoding delay and, hence, achieves high data rate at small n. Therefore, any choices of t are acceptable even when $2t \geq n$. The maximum r achieved by the BPSS-S and the BSPS-S interfaces is 0.75 due to the code dependent constraints. When P_e was required at 10^{-15} or better, the r achieved by the two decoders for $m = 5$ RS codes was merely above 0.6.

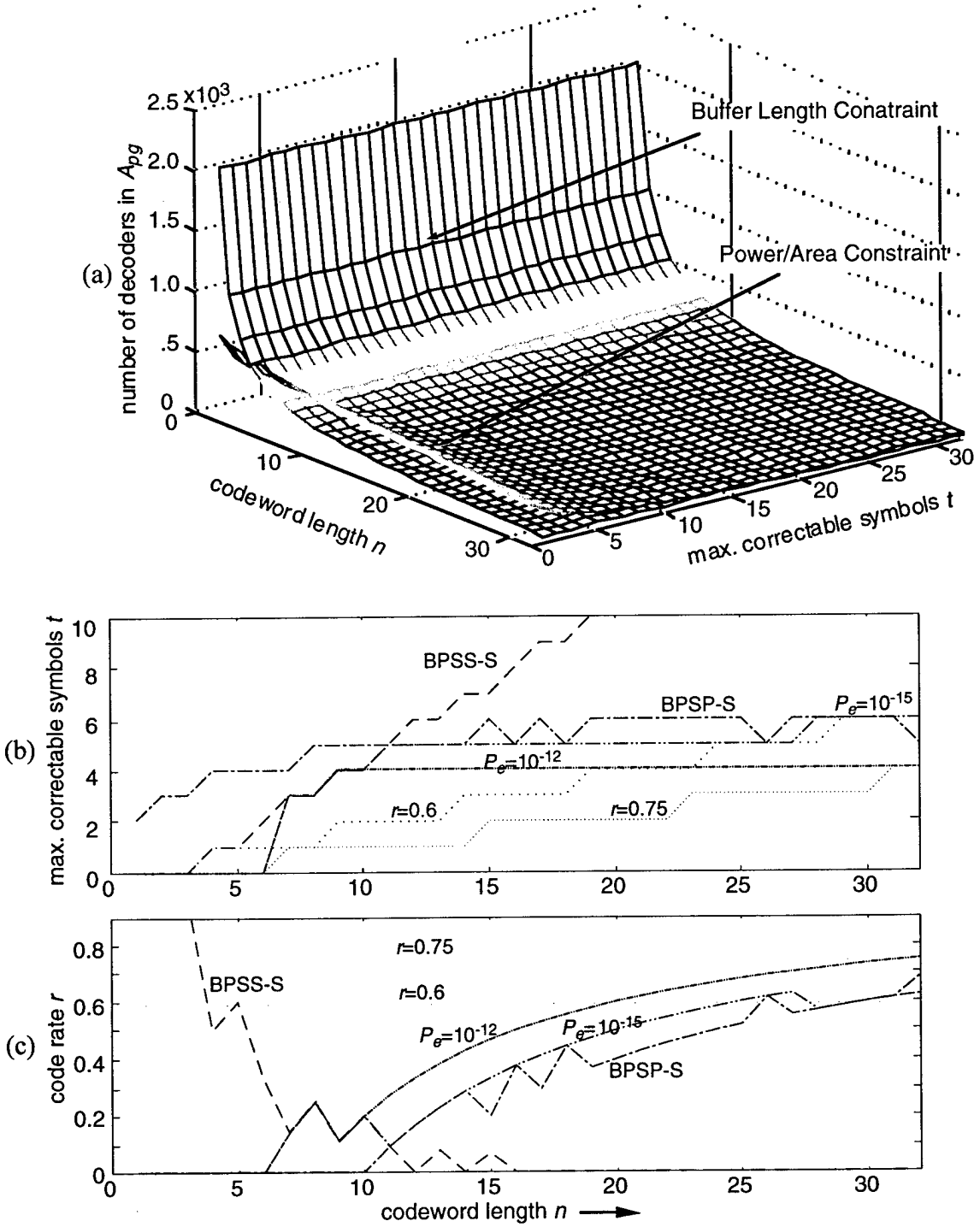

Figure 2.14 Feasibility analysis of the BPSS-S and BPSP-S decoders using the $m = 5$ RS codes. (a) the MINC (buffer length constraint) and MAXC (power/area constraint) of the BPSS-S interface; the BEC, CRC, MINC, and MAXC (b) on the n-t plane, and (c) the n-r plane ($F = 0.25$ μm, $P_b = 10\text{-}4$, $A_{pg} = 10$ cm2, $P_{pg} =$ 2 W/cm2, $N_{in} = 10_6$ bits, $t_a = 10$ μs, $f_c = 320$ MHz).

In order to achieve higher r, the RS codes of long n, e.g., 255, have to be used, and, as shown in Fig. 2.10, the BPSS-C is the only design that can implement such long RS codes. In the second example, the BPSS-C design and the $m = 8$ RS codes were analyzed using the result obtained from

the four constraints. Figure 2.15 shows the intersection of the MINC and MAXC in the *n-t* and *r-t* planes. The smallest n values that satisfy $P_e = 10^{-12}$ (10^{-15}) were obtained at 117 (150) which corresponds to $r = 0.897$ (0.893). The largest r for $P_e = 10^{-12}$ (10^{-15}) was obtained at $n = 237$ (256) with $r = 0.941$ (0.930) which is much higher than the r using the BPSS-S and BPSP-S designs for the $m = 5$ codes.

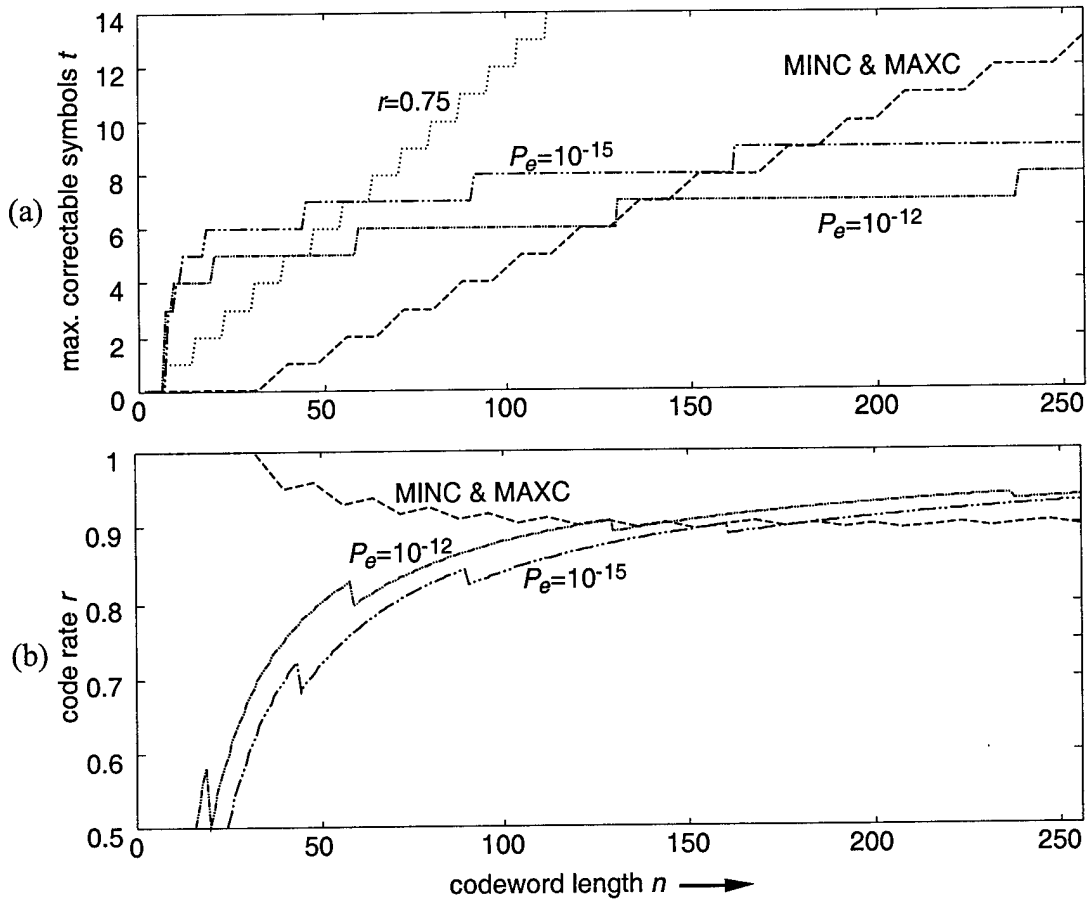

Figure 2.15 Feasibility analysis of the BPSS-C interfaces using the $m = 8$ RS codes. (a) BEC, CRC, MINC, and MAXC on the *n-t* plane, and (b) BEC, MINC, and MAXC on the *n-r* plane ($F = 0.25$ μm, $P_b = 10\text{-}4$, $A_{pg} = 10$ cm2, $P_{pg} = 2$ W/cm2, $N_{in} = 10^6$ bits, $t_a = 10$ μs, and $f_c = 320$ MHz).

2.7 Discussion and Conclusion

Optical page-oriented memory (OPOM), employing advanced photonic materials and optoelectronic devices, provides the large capacity and the high data access rate required by novel digital information applications. Unfortunately, uncoded OPOMs have a high raw bit error rate (BER) which presents a limitation. The use of error detection/correction is one way to reduce the BER to an acceptable level and improve overall memory capacity. Reed-Solomon (RS) codes are frequently used for error correction because they can effectively correct both random and burst errors. Likewise, RS codewords have a variety of lengths, and they are separated at the largest possible distance in the code space. We discussed the construction, specifications, and requirements of the output interface of OPOMs containing an array of RS decoders implemented using smart pixel (SP) technology. Each SP cell consists of an electrical RS decoder and an optical parallel I/O. Because of the large number of parallel I/O channels and a high processing rate, the

SP interface simultaneously reduces the BER to a desirable rate and provides a high aggregate data throughput. In this thesis, six SP implementations of the RS decoder using the transform decoding algorithm were analyzed to find the most effective implementation.

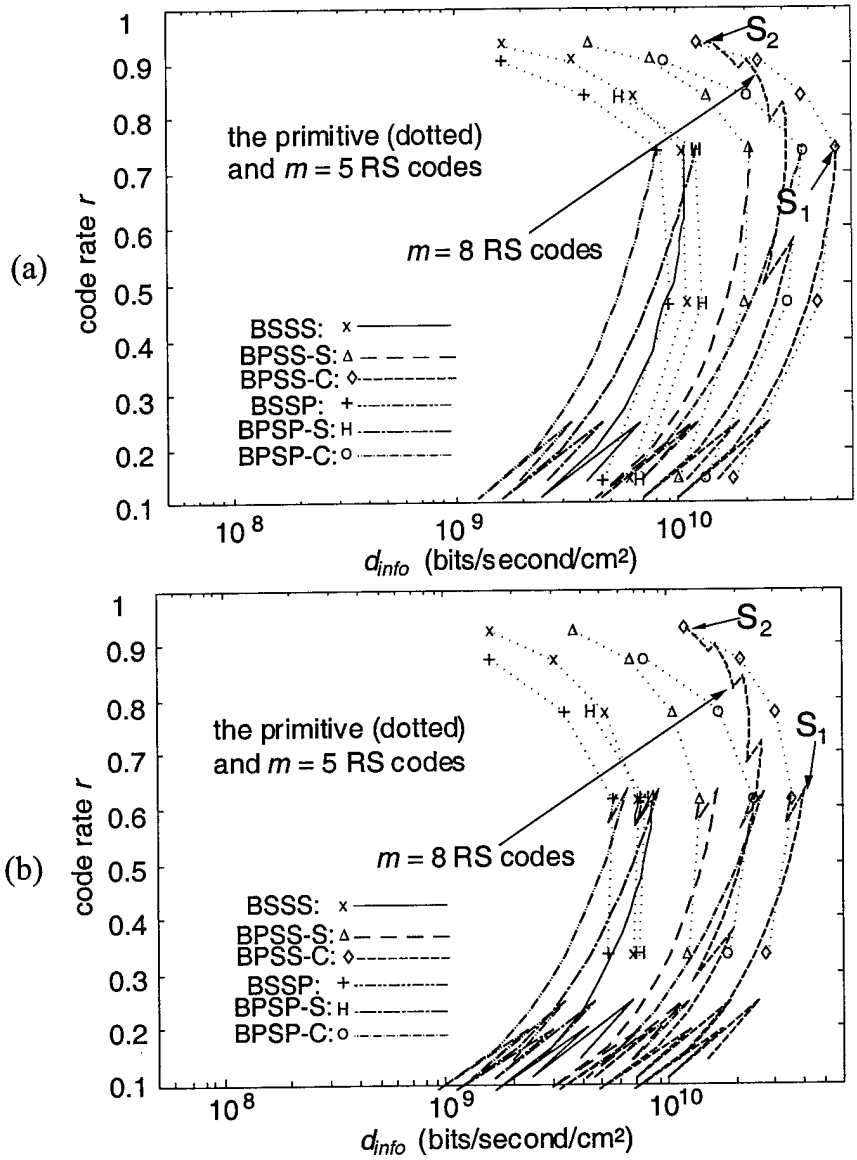

Figure 2.16 Relationship between the code rate r and the information spatial channel density d_{info} for (a) $P_e = 10^{-12}$ and (b) $P_e = 10^{-15}$. Point S1 (Section 5.B) denotes the highest d_{info}, S2 (Section 5.E) denotes the largest code rate, i.e., the largest usable storage capacity.

We summarize the results, as shown in section 5, of the performance of the six implementations in terms of: the input spatial channel density (d_{scin}); the aggregate input data throughput; and the information spatial channel density (d_{info}). d_{info} is optimized as functions of data page size, memory access time, and other physical conditions. It was shown that RS decoding processes need smart pixel technology to provide a large number of I/O because the d_{scin} of most implementations exceeds electrical limits, as shown in Fig. 2.9. The BPSS-C decoder for the (32, 24) RS code in GF(2^5) provides the largest d_{info} for an SP interface where $P_e = 10^{-12}$. In an SP interface where P_e

$= 10^{-15}$ (not shown here), the (27, 17) RS code also implemented by the BPSS-C decoder provides the largest d_{info}.

One of the objectives is to determine the largest memory capacity by optimizing the code rate of RS codes when the maximal data access rate, specified by the access time and the size of a data page, is known. It is achieved by determining RS codes which can satisfy the four following constraints: the bit error requirement; the code rate requirement; the buffer length limitation; and the VLSI power/area limitation. From the codes that meet these requirements, the code of the highest code rate is then selected, because such a code would utilize the memory capacity most effectively. For the example discussed, the (237, 223) RS code (r = 0.941) in GF(2^8) implemented by the BPSS-C design was the best selection when $P_e = 10^{-12}$. The (256, 238) RS code provided the highest r (= 0.930) when $P_e = 10^{-15}$. These results assume that page access time = 10 μs and page size = 1,024 × 1,024 bits.

Figure 2.16 shows the relationship between code rate r and d_{info} for $P_e = 10^{-12}$ and 10^{-15}. Section 5.B determines the RS code denoted by S_1 which represents the highest d_{info}, and Section 5.E determines S_2 representing the largest r. Note that r is proportional to the usable capacity of the OPOMs. The zigzags of the lines of fixed m come from discontinuities of r for various n at a fixed m, as shown in Fig. 2.4. When physical conditions are changed to increase the data throughput (e.g., smaller VLSI feature size, larger area, and larger power density), these lines move to the right without changing r.

From Fig. 2.16, the RS codeword length n tends to approach two extremes: achieving either high data throughput (shorter n), or large capacity (longer n). One possible way to extend the envelops of these conflicting requirements is to use 3-D VLSI packaging to implement long-length RS decoders. The 3-D packaging technique connects the multiple stacked substrates with electrical circuitry through optical vias. Individual modules of the TDA decoder are fabricated on separated substrates, and the substrates are aligned and interconnected to perform the pipelined decoding scheme. Another possibility is the product codes in which two RS codes with shorter n are combined. The product codes provide higher combined code rate than a regular RS code of the same error-correcting capability. In addition, the design of decoder for short-length codes is easier.

There are two other results discovered in this study. First, the VLSI circuit simulation model, SUSPENS, was originally developed for electronic general-purpose microprocessor chips. It was modified for the SP decoder array so that the buffers were estimated separately from the decoding logic and more proper parameters were used. However, two intuitive problems exist. First, the average number of transistors per logical gate is 50% larger than the average number of the general-purpose chips which might affect the use of the modified model. Secondly, the Rent's rule was obtained empirically from electronic circuitry where the pins are on the edge of a chip. For optoelectronic SP devices, the optical sources and receivers can be mounted together with the electronic components. This planar arrangement also affects the application of the modified SUSPENS to the estimation of SP devices.

The second result is that the TDA is not a 'good' scheme for shortened RS codes, nor for the RS codes with short n. As shown in Figs. 2.6, 2.7, and 2.11, the decoding hardware and power dissipation changed slightly as n decreases for a fixed m. In the implementations of the TDA, the ITES module needs many more logical gates than the other modules, and the number of logical gates is proportional to $m(N - 2t)$, where $N = 2^m - 1$. In order to achieve better performance design, different decoding schemes will be studied and applied.

Figure 2.17 shows the performance of the implementations studied in this thesis. The horizontal axis shows the input spatial channel density, and the vertical axis shows the information rate per channel. The three dashed lines represent the information spatial channel density at 10^{-6}, 10^{-9}, and 10^{-12} bits per second per cm^2, respectively. The d_{info} of the implementations of the SP interface for a large number of RS codes which reduce the BER for 10^{-4} to 10^{-12} and 10^{-15} is in the range of 10^8 to 10^{11} bits/second/cm^2. This results in the aggregate information rate up to 1 terabit per second in 10 cm^2. A parallel RS decoder is also shown [50].

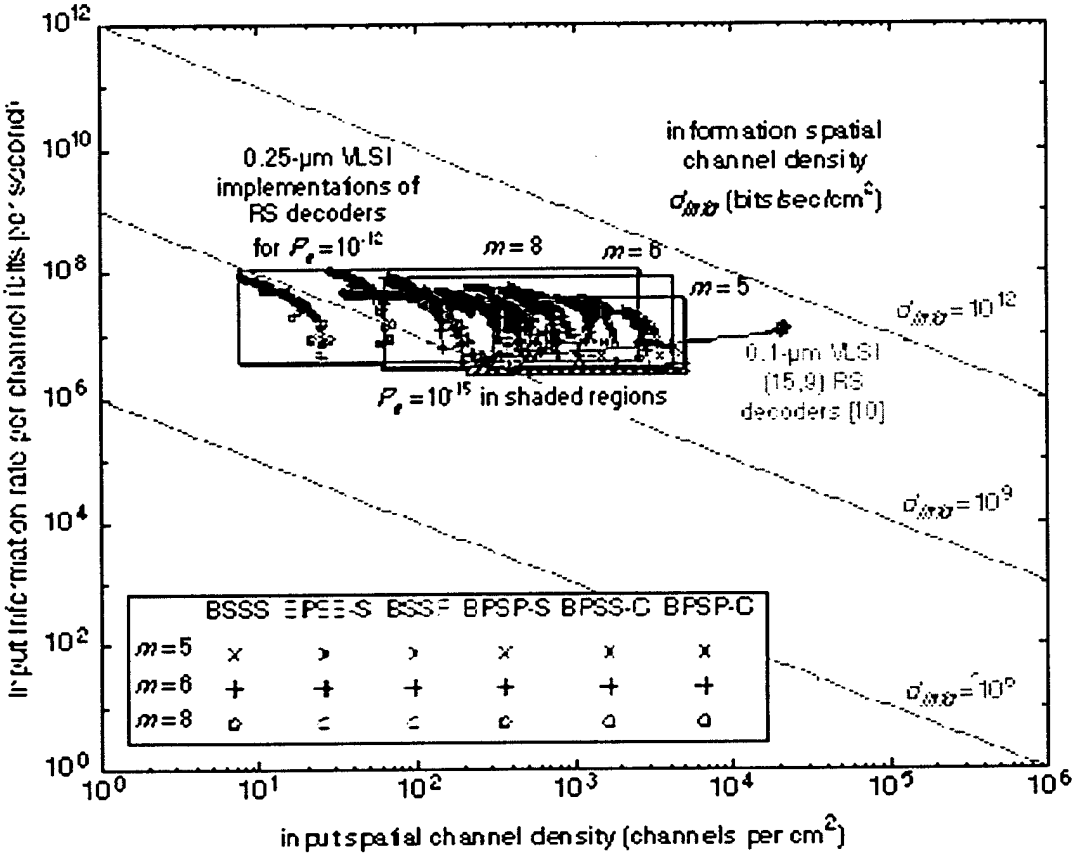

Figure 2.17 Performance of the optoelectronic smart pixel pixel error-correcting interfaces for optical page-oriented memories.

Appendix A. Modified SUSPENS: A VLSI Circuit Simulation Model

In order to estimate the performance of various circuitry in the SP interface, a VLSI circuit simulator is used. This circuit model originates from the SUSPENS [51] and is modified using Liu and Svensson's model [52]. The SUSPENS model is a system-level circuit simulator for central processing units (CPUs). The model estimates the clock frequency, power dissipation, and chip/module sizes of general-purpose processors by emphasis on the interactions among devices, circuits, logic, packaging, and architecture. In order to apply the SUSPENS model to the TDA decoder which is a special-purpose processor, we modified it using Liu and Svensson's model in which the power dissipated in clock distribution is taken into account. In addition, the on-chip SRAM and the logic gates are estimated using different sets of parameters. The TDA decoder does not have on-chip SRAM; instead, there are a large number of shift registers which have very different properties from the decoding logic gates. Therefore, two sets of parameters and formulas are used by the modified model to predict the TDA decoders.

In the modified circuit model, first, an upper limit to average wire length $\overline{R}$ (in units of gate pitch) is defined by

$$\overline{R} = \begin{cases} \frac{2}{9}\left(7\frac{N_g^{p-0.5}-1}{4^{p-0.5}-1} - \frac{1-N_g^{p-1.5}}{1-4^{p-1.5}}\right)\cdot\frac{1-4^{p-1}}{1-N_g^{p-1}}, & p \neq 0.5 \\ \frac{2}{9}\left(7\log_4 N_g - \frac{1-N_g^{p-1.5}}{1-4^{p-1.5}}\right)\cdot\frac{1-4^{p-1}}{1-N_g^{p-1}}, & p = 0.5. \end{cases} \quad \text{(A.1)}$$

It is obtained by applying Rent's rule to calculate the number of interconnections in a circuit block. Here, N_g is the number of logic gates in the block and p is the empirical Rent's constant for on-chip interconnection length calculation. In Eq. (A.1), Rent's constant p has been modified from the p calculated from Rent's rule, for example, $p = 0.4$ rather than $p = 0.6$ - 0.7 which is obtained directly from circuit layouts. On the other hand, the theoretical p is used when a factor of 0.54 is applied to the computed $\overline{R}$ [53], i.e.,

$$\overline{R} \leftarrow 0.54\overline{R}(\text{real } p) \quad \text{(A.2)}$$

Then, the average interconnection length in actual units is

$$l_{av} = \overline{R} \cdot d_g, \quad \text{(A.3)}$$

where d_g is the logic gate dimension in, for example, microns.

The logic gate dimension is limited by transistors when all gates can be placed right next to each other. In TDA decoders, the shift registers used for pipelining and buffers are transistor packing density limited. The logic gate dimension is computed as

$$d_{gtrlim} = \sqrt{k_g} \cdot F, \quad \text{(A.4)}$$

where k_g is a proportionality constant between gate area and F, the minimum feature size of CMOS technology used. In our simulation, $k_g = 67$ for a D-flip-flop consisting of 16 transistors.

Another case of the logic gate dimension happens in logic-intensive chips where area is normally limited by wiring capacity. In TDA decoders, the finite field multipliers, mod-2 adders, and other logic are characterized into the interconnection-capacity limit, and the logic gate dimension is given as

$$d_{gintlim} = \frac{f_g \overline{R} p_w}{e_w n_w}, \quad \text{(A.5)}$$

where f_g is the fan-out of a typical gate, p_w is wiring pitch, e_w is wiring efficiency, and n_w is the number of wiring levels. In our simulations, p_w is chosen as 3 times F although a larger factor, for examples, 4 or 5, is usually used in sub-micron VLSI technology. The wiring efficiency e_w is typically 0.4, and the wiring level n_w is assumed to be 3.

The logic gate dimension is the maximum of d_{gtrlim} and $d_{gintlim}$, i.e.,

$$d_g = \max\left(d_{gtrlim}, d_{gintlim}\right). \quad \text{(A.6)}$$

The dimension and area of the VLSI chip are then

$$D_c = d_g\sqrt{N_g}\,, \text{ and} \tag{A.7}$$

$$A_c = D_c^2 = d_g^2 N_g\,, \tag{A.8}$$

respectively.

The maximum clock frequency that can be achieved is estimated by

$$f_{c,max} = \left(f_{ld} T_g + R_{int} C_{int} \frac{D_c^2}{2} + \frac{D_c}{v_c} \right)^{-1}, \tag{A.9}$$

where f_{ld} is the logic depth, T_g is the average gate delay, v_c is the light speed, and R_{int} and C_{int} are the wiring resistance and capacitance per unit length, respectively. Due to the pipeline design of the TDA decoders and multipliers, the logic depth is much shorter (e.g., 4 to 8) than most general-purpose computing processors (from 8 to 30), which implies high operating speed for the TDA decoders.

The total power dissipation is estimated as

$$P_c = \frac{1}{2} f_c f_d N_g f_g \left(l_{av} C_{int} + 3 k_{tr} C_{tr} \right) V_{DD}^2 + \frac{1}{3}\frac{1}{2} N_p f_c f_d C_{out} V_{DD}^2 + f_c C_{totalclk} V_{DD}^2\,, \tag{A.10}$$

where f_c is the clock frequency used, f_d is the duty factor, C_{tr} is the capacitance of the minimum-size transistor, k_{tr} is the width/length (W/L) ratio of VLSI transistor, N_p is the number of inputs/outputs, C_{out} is the total capacitance at an output pin (= 50 pF), $C_{totalclk}$ is the total capacitance of clock distribution, and V_{DD} is the supply voltage. The first and second terms estimates the power consumed in performing logic functions and in the input/output buffers, respectively. The third term accounts for the power consumption in clock distribution. The total capacitance of clock distribution $C_{totalclk}$ is given by

$$C_{totalclk} = \left(1 + k_{driver}\right)\left(L_{clk} N_g C_{tr} + C_{clkwire} \right), \tag{A.11}$$

where k_{driver} the clock driver ratio (= 0.3), L_{clk} is the number of clock driven transistors in a logic gate, and $C_{clkwire}$ is the global clock wire capacitance and is approximated by

$$C_{clkwire} = 24 C_{int} D_c\,. \tag{A.12}$$

The original Eq. (A.10) in [47] contains a term of SRAM capacitance which is not used in TDA decoders, and then is omitted here. Finally, in a TDA decoder, the total power consumption P_c can be expressed as a combination of powers consumed in the logic part P_{logic}, buffers P_{buf}, input/output buffers P_{io}, and clock distribution P_{clk}, i.e.,

$$P_c = P_{logic} + P_{buf} + P_{io} + P_{clk}\,, \tag{A.13.a}$$

where

$$P_{logic} = \frac{1}{2} f_c f_d N_{logic} \left(\frac{d_{gintlim}^2 n_w e_w C_{int}}{p_w} + 3 k_{tr} f_g C_{tr} \right) V_{DD}^2, \quad \text{(A.13.b)}$$

$$P_{buf} = \frac{1}{2} f_c f_d f_g N_{buf} \left(d_{gtrlim} \overline{R} C_{int} + 3 k_{tr} C_{tr} \right) V_{DD}^2, \quad \text{(A.13.c)}$$

$$P_{io} = \frac{1}{3} \frac{1}{2} N_p f_c f_d C_{out} V_{DD}^2, \text{ and} \quad \text{(A.13.d)}$$

$$P_{clk} = f_c C_{totalclk} V_{DD}^2. \quad \text{(A.13.e)}$$

Here, N_{logic} and N_{buf} are the number of logic gates of the logic and buffer circuits, and $N_g = N_{logic} + N_{buf}$.

2.8 References

[1] J. F. Heanue, M. C. Bashaw, and L. Hesselink, "Volume holographic storage and retrieval of digital data," *Science* **265**, 749-752 (1994).

[2] H. Ishio, "Next-generation communications networks and optical fiber technologies," *Optoelectronics-Devices and Technologies* **10**, 3-14 (1995); and Special Issue on Optical Interconnections for Information Processing, *IEEE J. Lightwave Technol.* December 1995.

[3] D. Chen and J. D. Zook, "An overview of optical data storage technology," *Proc. IEEE* **63**, 1207-1230 (1975).

[4] T. Parish, "Crystal clear storage," *Byte*, 283-288, November 1990.

[5] S. Jutamulia and G. M. Storti, "Three-dimensional optical digital memory," *Optoelectronics-Devices and Technologies* **10**, 343-360 (1995).

[6] M. A. Neifeld and M. McDonald, "Error correction for increasing the usable capacity of photorefractive memories," *Opt. Lett.* **19**, 1483-1485 (1994).

[7] T. R. N. Rao and E. Fujiwara, *Error-Control Coding for Computer Systems*, Prentice-Hall, Englewood Cliffs, New Jersey, 1989.

[8] S. R. Whitaker, J. A. Canaris, and K. B. Cameron, "Reed Solomon VLSI codec for advanced television," *IEEE Trans. Circuits and Systems for Video Technol.* **1**, 230-236 (1991).

[9] M. A. Neifeld and J. D. Hayes, "Parallel error correction for optical memories," *Opti. Mem. Neur. Netw.* **3**, 87-98 (1994).

[10] M. A. Neifeld and J. D. Hayes, "Error-correction schemes for volume optical memories," *Appl. Opt.* **34**, 8183-8191 (1995).

[11] A.A. Sawchuk, "Smart Pixel Devices and Free-Space Digital Optics Applications," *LEOS '95 Conference Proceedings,* IEEE Lasers and Electro-Optics Society, 1995 Annual Meeting, San Francisco, November 1995, pp. 268-269, (invited paper).

[12] H. M. Shao, T. K. Trung, L. J. Deutsch, J. H. Yuen, and I. S. Reed, "A VLSI design of a pipeline Reed-Solomon Decoder," *IEEE Trans. Comput.* **C-34**, 393-403 (1985).

[13] P. J. van Heerden, "Theory of optical information storage in solids," *Appl. Opt.* **2**, 393-400 (1963).

[14] S. Boj, G. Pauliat, and G. Roosen, "Dynamic holographic memory showing readout, refreshing, and updating capabilities," *Opt. Lett.* **17**, 438-410 (1993).

[15] L. Hesselink and M. C. Bashaw, "Optical memories implemented with photorefractive media," *Opt. Quantum Electron.* **25**, 611-661 (1993).

[16] F. H. Mok, M. G. Tackitt, and H. M. Stoll, "Storage of 500 high-resolution holograms in a $LiNbO_3$ crystal," *Opt. Lett.* **16**, 605-607 (1991).

[17] K. Rastani, "Storage capacity and cross talk in angularly multiplexed holograms: two case studies," *Appl. Opt.* **32**, 3772-3778 (1993).

[18] C. Gu, J. Hong, I. McMichael, R. Saxena, and F. Mok, "Cross-talk-limited storage capacity of volume holographic memory," *J. Opt. Soc. Am. A* **9**, 1978-1983 (1992).

[19] J. F. Heanue, M. C. Bashaw, and L. Hesselink, "Sparse selection of reference beams for wavelength- and angular-multiplexed volume holography," *J. Opt. Soc. Am. A* **12**, 1671-1676 (1995).

[20] G. A. Rakuljic, V. Leyva, and A. Yariv, "Optical data storage by using orthogonal wavelength-multiplexed volume holograms," *Opt. Lett.* **17**, 1471-1473 (1992).

[21] S. Yin, H. Zhou, F. Zhao, M. Wen, Z. Yang, and F. T. S. Yu, "Wavelength multiplexed holographic storage in a sensitive photorefractive crystal using a visible-light tunable diode laser," *Opt. Comm.* **101**, 317-321 (1993).

[22] H. Sasaki, J. Ma, Y. Fainman, S. H. Lee, and Y. Taketomi, "Fast update of dynamic photorefractive optical memory," *Opt. Lett.* **17**, 1468-1470 (1992).

[23] C. Denz, G. Pauliat, and G. Roosen, "Volume hologram multiplexing using a deterministic phase encoding method," *Opt. Comm.* **85**, 171-176 (1991).

[24] D. Psaltis, M. Levene, A. Pu, G. Barbastathis, and K. Curtis, "Holographic storage using shift multiplexing," *Opt. Lett.* **20**, 782-784 (1995).

[25] D. Brady and D. Psaltis, "Control of volume holograms," *J. Opt. Soc. Am. A* **9**, 1167-1182 (1992).

[26] K. Curtis, A. Pu, and D. Psaltis, "Method for holographic storage using peristrophic multiplexing," *Opt. Lett.* **19**, 993-994 (1994).

[27] L. Hesselink and S. Redfield, "Photorefractive holographic recording in strontium barium niobate fibers," *Opt. Lett.* **13**, 877-879 (1988).

[28] F. H. Mok, "Angle-multiplexed storage of 5000 holograms in lithium niobate," *Opt. Lett.* **18**, 915-917 (1993).

[29] D. Psaltis and A. Pu, "Holographic 3-D disks," *Optoelectronics-Devices and Technologies* **10**, 333-342 (1995).

[30] D. A. Parthenopoulos and P. M. Rentzepis, "Three-dimensional optical storage memory," *Scinece* **245**, 843-845 (1989).

[31] S. Hunter, F. Kiamilev, S. Esener, D. A. Parthenopoulos, and P. M. Rentzepis, "Potentials of two-photon based 3-D optical memories for high performance computing," *Appl. Opt.* **29**, 2058-2066 (1990).

[32] C. De Caro, A. Renn, and U. P. Wild, "Hole burning, Stark effect, and data storage: 2: holographic recording and detection of spectral holes," *Appl. Opt.* **30**, 2890-2898 (1991).

[33] H.-J. Muschenborn and U. P. Wild, "holographic image storage and molecular computing using spectral hole-burning," *Optoelectronics-Devices and Technologies* **10**, 311-332 (1995).

[34] M.-P. Bernal, H. Coufal, R. K. Grygier, J. A. Hoffnagle, C. M. Jefferson, R. M. Macfarlane, R. M. Shelby, G. T. Sincerbox, P. Wimmer, and G. Wittmann, "A precision tester for studies of holographic optical storage materials and recording physics," submitted to *Applied Optics*.

[35] A. Pu and D. Psaltis, "High density recording in photopolymer-based holographic 3-D disks," to be published in *Applied Optics*, May 1996.

[36] D. A. B. Miller, "Quantum-well self-electro-optic effect devices," *Opt. Quantum Electron.* **22**, S61-S98 (1990).

[37] A. L. Lentine, *et al.*, "Field-effect-transistor self-electro-optic effect device (FET-SEED) electrically addressed differential modulator array," *Appl. Opt.* **33**, 2849-2855 (1994).

[38] A. V. Krishnamoorthy, *et al.*, "3-D integration of MQW modulators over active submicron CMOS circuits: 375 Mb/s transimpedance receiver-transmitter circuit," *IEEE Photon. Technol. Lett.* **7**, 1288-1290 (1995).

[39] D. J. McKnight, K. M. Johnson, and R. A. Serati, "256 × 256 liquid-crystal-on-silicon spatial light modulator," *Appl. Opt.* **33**, 2775-2784 (1994).

[40] A. Ersen, S. Krishnakumar, V. Ozguz, J. Wang, C. Fan, S. Esener, and S. H. Lee, "Design issues and development of monolithic silicon/lead lanthanum zirconate titanate integration technologies for smart spatial light modulators," *Appl. Opt.* **31**, 3950-2965 (1992).

[41] K. Kasahara, "VSTEP-based smart pixels," *IEEE J. Quantum Electron.* **29**, 757-768 (1993).

[42] J. L. Jewell, Y. H. Lee, A. Scherer, S. L. McCall, N. A. Olsson, J. P. Harbison, and L. T. Florez, "Surface-emitting microlasers for photonic switching and interchip connections," *Opt. Eng.* **29**, 210-214 (1990).

[43] M. Hibbs-Brenner, S. Mukherjee, J. Skogen, B. Grung, E. Kalweit, and M. Bendett, "Design, fabrication and performance of an integrated optoelectronic cellular array," *Proc. Optical Enhancements to Computing Technology, SPIE* **1563**, 10-20 (1991).

[44] J. J. Brown, J. T. Gardner, and S. R. Forrest, "An integrated optically powered, optoelectronic "smart" logic pixel for interconnection and computing applications," *IEEE J. Quantum Electron.* **29**, 715-726 (1993).

[45] S. B. Wicker and V. K. Bhargava, Ed., *Reed-Solomon Codes and Their Applications*, IEEE Press, New York, 1994.

[46] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error Correcting Codes*, North-Holland, Amsterdam, 1977.

[47] S. Lin and D. J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*, Prentice-Hall, Englewood Cliffs, New Jersey, 1983.

[48] G. C. Clark, Jr. and J. B. Cain, *Error-Correction Coding for Digital Communications*, Plenum Press, New York, 1981.

[49] H. Imai, Ed., *Essentials of Error-Control Coding Techniques*, Academic Press, New York, 1990.

[50] S. K. Sridharan and M. A. Neifeld, "Parallel error correction for page access optical memories," *OSA Annual Meeting*, TuE5 (Post deadline paper), Portland, September 10-15, 1995.

[51] H. B. Bakoglu, *Circuits, Interconnections, and Packaging for VLSI*, (VLSI Systems Series), Addison Wesley, Reading, MA, 1990.

[52] D. Liu and C. Svensson, "Power consumption estimation in CMOS VLSI chips," *IEEE J. Solid -State Circuits* **29**, 663-670 (1994).

[53] W. E. Donath, "Placement and average interconnection lengths of computer logic," *IEEE Trans. Circuit and Systems* **CAS-26**, 272-277 (1979).

3.0 Wavelength-Division-Multiplexing for High-Speed Network Gateways, Alan Willner

3.1 Introduction

Wavelength-division multiplexing (WDM), in which many wavelength-specific channels are simultaneously transmitted along the same optical fiber, has the potential for dramatically increasing the aggregate system capacity of high-speed networks. Additionally, each wavelength can represent the communications link being established between source and destination in a large network, thus enabling highly-efficient **wavelength-dependent data-packet routing**.

In a simple sense, N wavelengths can accommodate N different users. However, several technological issues will probably limit the total number of wavelengths in a network to <50. One scheme for enabling a large WDM network is to allow wavelength re-use, in which the same wavelength determines a different path at distinct parts of the network. Such wavelength re-use would require signal wavelength shifting in which a data signal traversing a large network must be periodically routed onto different available wavelengths thereby enabling it to reach the final destination. It is highly desirable to perform wavelength shifting all-optically to maintain high system speed and throughput. All-optical wavelength shifting of a data packet can be performed in a straightforward manner by utilizing the fast (<1 ns) gain properties of a semiconductor optical amplifier (SOA).

The operation of an all-optical wavelength shifter must include critical **routing** issues which must be addressed when operating high-speed optical networks over wide and local areas. *We are concerned with the utilization of photonic technology for data-fusion networks.* Some of these issues include:

(i) the demonstration of wavelength routing by using the control information encoded in a multiple-pilot-tone subcarrier header. Header detection, header removal, packet gating and wavelength shifting are all performed by a single multifunctional SOA. Based on the header information contained in the 60-ns-long pilot tones, each incoming 1 Gb/s data packet either: (i) passes through the switch unaffected, or (ii) is wavelength shifted and dropped at the switch. ***(Project 3.2.1)***

(ii) the demonstration of using optical buffering and wavelength shifting to accommodate rapid resolution of output port contention ***(Project 3.2.2)***

(ii) the use of a semiconductor optical amplifier to simultaneously and independently wavelength shift ***multiple*** input channels based on temporal multiplexing and spatial multiplexing. ***(Projects 3.2.3 and 3.2.4)***

(ii) the demonstration of all-optical conversions between the RZ and NRZ data formats which leads to format transparent WDM switching nodes. ***(Project 3.2.5)***

(iii) a method for ensuring polarization insensitivity and high output contrast ratio in an SOA-based wavelength shifter. ***(Project 3.2.6)***

We have attacked these issues by attempting to integrate WDM at the **gateway interfaces** between local and regional networks, between regional and global networks, and at each switching node itself. The projects described below will help enable a functional all-optical high-speed data fusion network.

3.2 Research Progress

We provide a brief summary of each of the projects supported under the FRI program from 9/96 till 9/97.

3.2.1 A Wavelength-Routing Node Using Multifunctional Semiconductor Optical Amplifiers and Multiple-Pilot-Tone-Coded Subcarrier Control Headers

Wavelength-division-multiplexing (WDM) can dramatically increase the capacity of optical networks by: (i) simultaneously of multiple channels on the same fiber located at different wavelengths, and (ii) providing wavelength-dependent routing paths through the network. However, due to several limitations in the total number of wavelengths available in a large network, it may be advantageous to provide for reuse of a limited set of wavelengths throughout a WDM optical network by means of all-optical wavelength shifting at key switching nodes or gateways. As an example of the potential need for wavelength shifting, a WDM wide-area network (WAN) may be composed of many WDM local-area networks (LAN), with certain wavelengths accessing the WAN and other wavelengths accessing each LAN (or individual node). For a WDM access node at the WAN-LAN gateway, a wavelength routing function is required. For instance, if data packets are destined for a given LAN, they will be spatially dropped (i.e., routed) to a given switch output port for that LAN and wavelength shifted to the appropriate destination node wavelength; however, if the data is not destined for this LAN, it will pass through the access node unaffected (see Fig 3.1 (a)). This wavelength and/or space routing decision must be performed at this WDM node for each packet, potentially requiring costly Gb/s high speed electronics. The use of subcarrier header control has been proposed and demonstrated as a possible solution to this problem, with the following advantages: (i) the data speed on the control subcarrier can be much lower than the data packet bit rate (i.e. <100 Mb/s), (ii) a single photodetector (connected to a bank of RF filters) can recover many different RF control signals located on several wavelengths whereas several photodetectors would be necessary to recover baseband control from several different WDM channels, (iii) the RF subcarrier technology is relatively mature and relatively cost effective, and (iv) the header and baseband are sharing the same wavelength (as opposed to different wavelengths) and can co-propagate with other wavelengths in the same fiber without incurring dispersion-induced walk-off between packet and header or wasting valuable available wavelengths. Recent work has reported the dynamic wavelength shifting by using an 8-bit 50-Mbit/s-modulated subcarrier control header.

An SOA has previously been investigated as a multifunctional device, e.g., a simultaneous channels dropper and channel adder and a simultaneous channel dropper and wavelength shifter. In this letter, we report an experimental demonstration of dynamic space and wavelength routing based on multifunctional SOAs for which the following functions are performed simultaneously: (1) header detection, (2) header removal, (3) packet gating, and (4) wavelength shifting. Additionally, a multiple-pilot-tone-coded subcarrier header scheme is proposed and implemented to reduce the processing and transmission delay as well as increase the number of available WDM network addresses.

In our multiple-pilot-tone-coded subcarrier header scheme for which each of n different subcarrier tones have m header bits each, the number of addressable nodes would be 2^{m*n} (see Fig. 3.1 (b)). This scheme differs from that reported since the number of required subcarriers in our scheme scales logarithmically with the number of addressable nodes, not linearly. It may also be a desirable alternative to a multiple-wavelength-coded header scheme because dispersion will limit the number of parallel wavelengths which can be used for addressing when the distance of the optical path is not small.

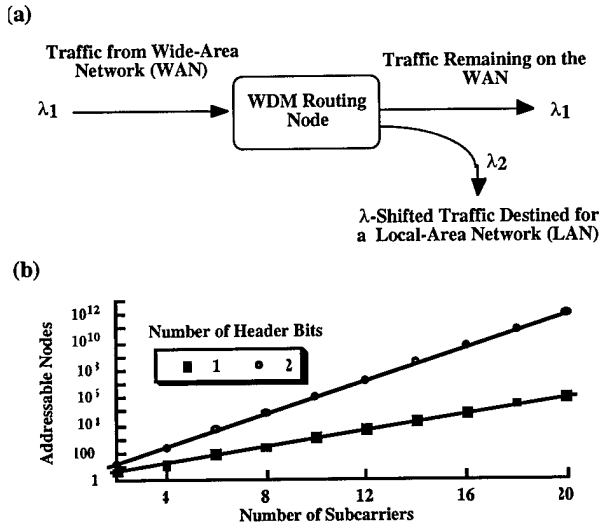

Figure 3.1 (a) Conceptual diagram of a WAN-LAN wavelength routing node. (b) Addressable capacity as a function of the number of coding subcarriers.

Figure 3.2 shows the experimental setup. The incoming packets are located at 1571 nm (λ_1). The baseband is ASK modulated at 1 Gb/s and is preceded by a multiple-pilot-tone subcarrier header. Each packet is 640-ns long, consisting of: a 60-ns one-bit header, a 53-byte ATM 1 Gb/s baseband data stream, header-baseband guard time, and inter-packet guard time. The multiple-pilot-tone consists of three subcarriers f_1, f_2 and f_3 transmitted in parallel time slots located at 1.2, 1.4 , and 1.6 GHz, respectively. The optical modulation index are ~20 % for three subcarriers. The specific routing information is determined as follows: f_1 informs the switch as to whether the packets are destined for this LAN; f_2 and f_3 inform the switch as to which of the 4 local wavelengths the packet will be shifted onto. Since the dynamic routing to a specific local wavelength has already been demonstrated, we focus on the packet passing through the switch unaffected or dropped at the switch and wavelength shifted for a local node destination. Passing or dropping is determined by switching "on" and "off" f_1 while keeping f_2 and f_3 always "on". The packets are programmed so that 33% of the traffic is destined for this access node while 66% of the traffic is passed on to other nodes.

Multifunctionality of the SOA is demonstrated as follows. At the beginning of each packet time slot, the pilot-tone header is detected by the reverse biased SOA_1. The detected pilot-tones are tapped, high-pass filtered, and demodulated by 3 subcarrier demodulators and input in parallel to the control board. When f_1 is "on", the control board emits a signal to switch SOA_1 "on" while switching SOA_2 "off". The incoming packets will saturate the gain of SOA_1, and all data is inversely copied and wavelength shifted to a cw probe signal at 1552 nm (λ_2) through SOA cross-gain compression. When f_1 is "off", SOA_1 will be switched "off" and SOA_2 will be switched "on", allowing the packets to pass this access node unaffected. The switched signals are input into baseband packet selector (not shown) to strip the subcarrier header and then input to the bit-error-rate tester (BERT). The module, therefore, simultaneously performs both space and wavelength switching. When reversed biased, the SOA_1 detector efficiency is ~0.4 A/W at 1571 nm which is about half that of a commercially-available pin detector. This is primarily due to the 2-3 dB coupling loss incurred between the fiber

and the SOA. The input power to SOA_1 is -5.2 dBm for the 1571-nm wavelength and -13.8 dBm for the 1552-nm wavelength. The input power to SOA_2 is -13.5 dBm for the 1571-nm wavelength.

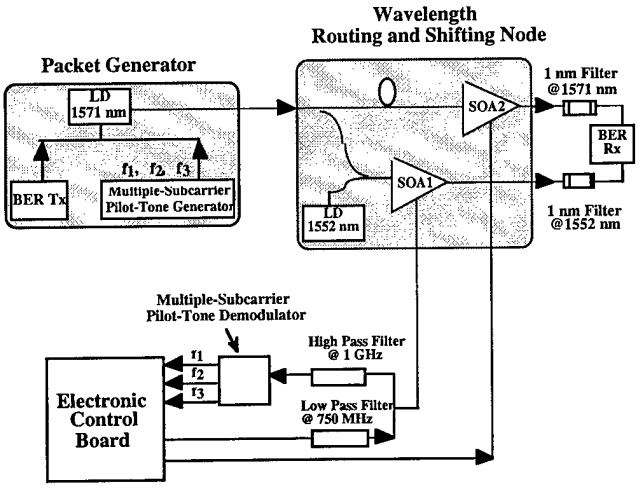

Fig. 3.2 Experimental setup for the wavelength routing node.

The real-time wavelength routing is shown in the oscilloscope traces of Fig. 3.3. During the first time slot, subcarrier f_1 is "on" causing this packet to be dropped to this node and wavelength shifted from 1571 to 1552 nm. Note that the old header has been stripped by switching "off" the SOA during the header detection. This may benefit the easier insertion of the new header because, as a response to the WAN-LAN switching, the header might need to be replaced or updated. Furthermore, the wavelength shifted signal incurs a non-severe contrast ratio degradation due to finite SOA gain saturation. In our experiment, the output extinction ratio is ~ 8 dB. During the second and third time slots, subcarrier f_1 is "off" and the packets pass straight through the node. The gradual decreasing signal levels of these two consecutive packets are caused by the electrical low-frequency cutoff of our bias-T for current input to SOA_2. Note that the addressing capacity can be enhanced by increasing the number of coding subcarriers or bits-per-subcarrier header. However, this capacity will eventually be limited by the sensitivity of recovering an individual subcarrier as well as the intermodulation between subcarriers.

Fig. 3.4 shows the recovered partial bit patterns for input packets and output packets being either dropped or passed straight through. As expected, the straight-through bit patterns are nearly identical to the input ones. The wavelength-shifted bit pattern is an inverse version of the input one because SOA cross gain compression method is used for the all-optical wavelength shifting. The bit-error-rate measurements were also taken. The sensitivity for passed packets and wavelength shifted packets are -29.0 dBm and -28.6 dBm respectively. Compared to the baseline curve, the power penalty for the passed packets is 1.4 dB and is mainly due to: (i) signal distortion by SOA_2, and (ii) additive amplified-spontaneous-emission from SOA_2. The power penalty for the wavelength shifted signal is 1.8 dB and is mainly due to the contrast ratio degradation upon wavelength conversion.

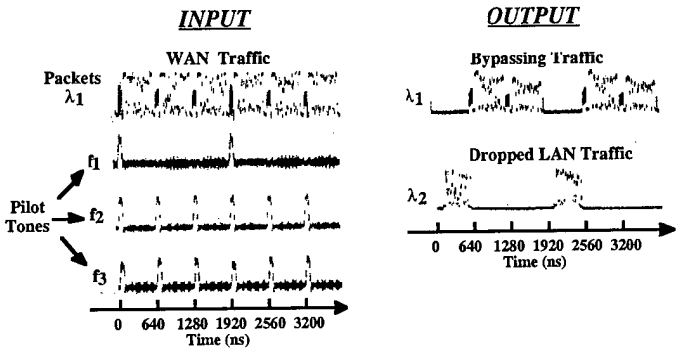

Fig. 3.3 Oscilloscope traces of input packets, output bypassed packets and output dropped packets. Also shown are the coding subcarrier pilot tones (f_1, f_2 and f_3).

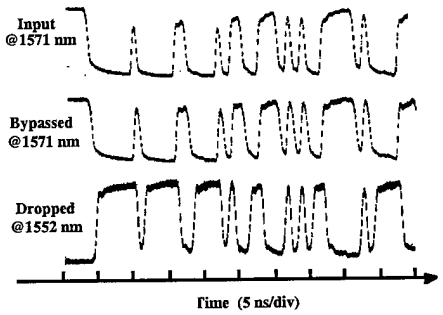

Fig. 3.4 Bit patterns of the input packets, output bypassed packets and output dropped packets (wavelength shifted).

3.2.2 Contention Resolution of High-Speed WDM Packets Using a Dynamically-Controlled Multiple-Wavelength Fiber Loop Buffer and Wavelength Shifting

Wavelength-division multiplexing (WDM) may enable highly functional optical networks in which wavelength is used to provide higher capacity and efficient routing of data to different destinations. Passive wavelength routing using integrated frequency routers may provide a desirable solution. However, a packet may require all-optical wavelength shifting at key dynamic network gateways in order to be routed to the appropriate destination in which: (i) passive wavelength routing is used, or (ii) wavelength re-use is employed due to an insufficient number of available wavelengths. For either reason, a packet destined for a given wavelength-dependent node may require optical buffering and all-optical wavelength shifting to find an open time slot on an appropriate wavelength. Such buffering and wavelength shifting provides a solution to a key challenge in efficient WDM networks, that being output-port contention resolution for which 2 input packets wish to be routed to the same destination on the same wavelength (Figure 3.5 (a)). Optical buffering of the contending packets may be essential to obtain a low cell-loss ratio, and these packets must be dynamically inserted into free time slots on the desired wavelength. Contention-resolution techniques in all-optical networks include: a series of delay lines, multiple-wavelength buffers, deflection routing, and 2x2 WDM switching nodes. Some of the inherent disadvantages of these methods, which we have addressed, include: (1) the series of delay lines must continue to grow with an increase in the number of buffer delay times and did not include dynamic wavelength shifting and network reconfigurability, (2) the multiple-wavelength buffer did not include control or self-routing and was only for a single one-packet delay, (3) deflection routing is not optimally efficient for a network, (4) the 2x2 WDM switching node did not incorporate any buffering if the desired wavelength was already in use.

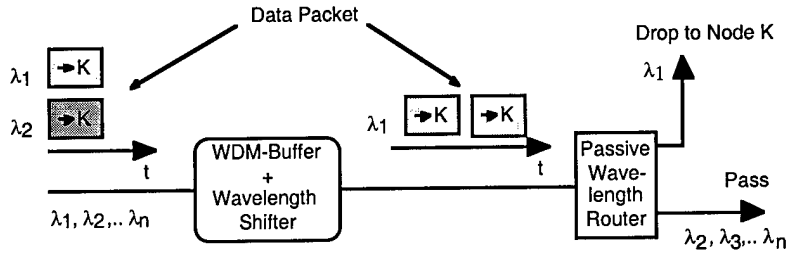

Fig. 3.5 (a) Conceptual diagram of add/drop multiplexing and wavelength routing requiring buffering and wavelength shifting.

We experimentally demonstrate dynamic contention resolution for a system in which self-routing packets on different 1 Gbit/s WDM channels compete for the same output wavelength channel. This function, important in a reconfigurable network which incorporates elements of passive wavelength routing, is realized using an electronically controlled multi-wavelength fiber loop buffer, high speed optical switches, and a single SOA based wavelength shifter. The added functionality of this design can accommodate several input wavelengths and can switch randomly to several output wavelengths over several buffer-enabled time slots. We demonstrate the buffering of one contending packet for one or two packet lengths, according to the detection of contention by reading the input packet headers. The stored packet is dynamically switched into the packet stream when a free time slot on the desired wavelength is detected. This packet is then all-optically shifted to the desired wavelength. Since high speed optical switches are used, only a short guard time is necessary. The dynamic buffering introduces a low power penalty of 1 dB from the space switch and 2.5 dB from the all-optical wavelength shifting, which is based on SOA cross-gain compression (XGC). Efficiency, throughput and functionality are enhanced by using this method of contention resolution.

The experimental setup is shown in Fig. 3.5 (b). There is one WDM input with 1 Gbit/s packets located at λ_1 (1557 nm) and λ_2 (1552 nm) with a packet length of 424 bits including a 16 bit header and a payload. Suitable guard bands are inserted to illustrate clearly. Note that our design can accommodate guard times on the order of a few ns, although we are limited by external gating. The use of $LiNbO_3$ high speed optical switches introduces some polarization sensitivity, which do not impact our 2 stage buffer and which could be reduced by polarization independent switches. Output-port wavelength channel contention is determined by: (i) tapping off some power of each wavelength channel, (ii) detecting the two packet headers, (iii) comparing one header information bit, and (iii) switching the packet at λ_2 into the fiber loop buffer if contention exist. When an empty slot is determined to be available, the buffered packet at λ_2 is switched out of the loop and wavelength shifted to λ_1 using XGC. Several packet-time-slot delays are possible.

The real-time contention resolution is shown in the oscilloscope traces of Fig. 3.6. For our demonstration, all data packets on the two WDM channels are routed to λ_1. In one case, each full input packet time slot is followed by an empty one. The λ_2 packet is successfully delayed by one packet length and then shifted onto λ_1. In the second case, only every third packet time slot at λ_1 is empty, requiring that the packets at λ_2 are experimentally delayed by two packet lengths. Note that even a few delay packet slots will still provide a significant decrease in packet dropping probability due to output-port contention.

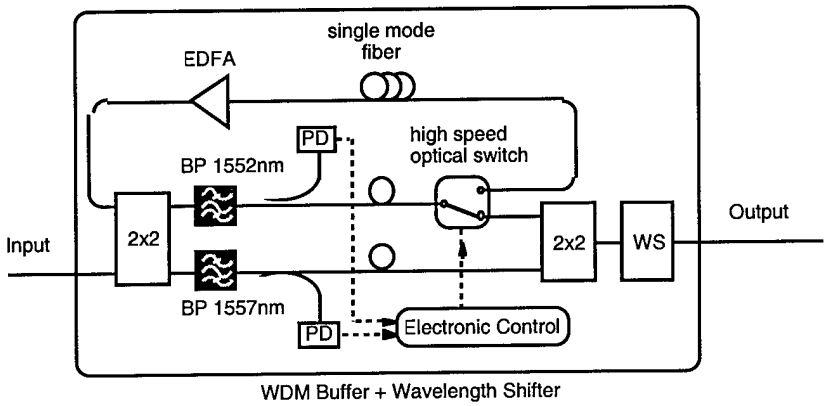

Fig. 3.5 (b) Experimental setup of the active fiber loop buffer (PD: Photodetector, BP: Bandpass Filter.)

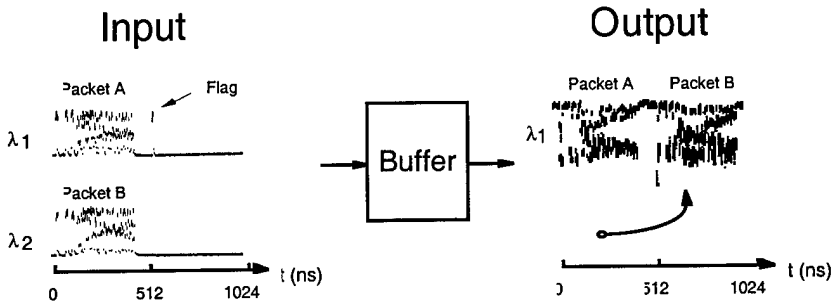

Fig. 3.6 (a) Oscilloscope trace of the inputs (λ_1 and λ_2) and output (λ_1) for one-packet-length delay.

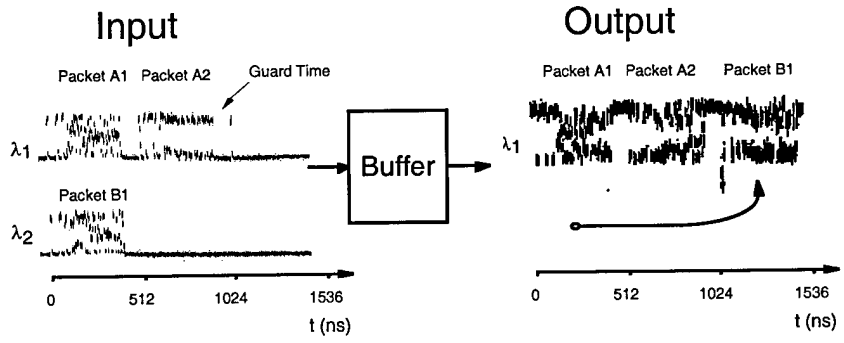

Fig. 3.6 (b) Oscilloscope trace of the inputs (λ_1 and λ_2) and output (λ_1) for two-packet-length delay.

3.2.3 Experimental Demonstration of a Multiple-_ Wavelength Shifter for Dynamically Reconfigurable WDM Networks

Future all-optical wavelength-division-multiplexed (WDM) networks may require the reuse of a finite set of available wavelengths in order to maximize throughput and efficiency. Such wavelength reuse can be achieved by wavelength shifting a given WDM channel's wavelength to that of a different available wavelength. Many wavelength shifting schemes have been demonstrated, and some have

even been able to simultaneously wavelength shift multiple input signals onto multiple output wavelengths by shifting a fixed block of wavelengths using four-wave-mixing. However, none have shown the capability to *independently* wavelength shift multiple input WDM channels to randomly different output wavelengths. We demonstrate a novel multiple-_ wavelength shifter that can simultaneously wavelength shift several WDM channels by utilizing: (i) semiconductor optical amplifier cross-gain compression, (ii) bit time interleaving, and (iii) appropriate gating. Furthermore, our wavelength shifter is *transparent* to both the NRZ and RZ input data-formats. We demonstrate our multiple-_ wavelength shifter by *simultaneously* shifting the wavelengths of *two* independent WDM channels from 1548 and 1552 nm to 1540 and 1569 nm, respectively, with low power penalties. Although we demonstrate our time-interleaved wavelength shifter for only two input WDM channels, its capacity can be extended to accommodate more channels.

Figure 3.7 conceptually shows a multiple-_ wavelength shifting module that takes two input WDM "tributaries" and shifts each channel's wavelength to that of a different available wavelength. The wavelength shifted signals are then routed to different destinations. Also shown in Figure 3.7 is a robust conceptual implementation of such a module that includes a data-format transparent front-end to the multiple-_ wavelength shifter. The purpose of this front-end is to make the operation of our multiple-_ wavelength shifting module transparent to both the NRZ and RZ input data-formats. The input WDM channels can both be either NRZ or RZ formatted or a combination of the two formats without affecting the performance of the system. The multiple-_ wavelength shifter then simultaneously shifts the input WDM tributaries' signal wavelengths to different available wavelengths, *all within a single device.*

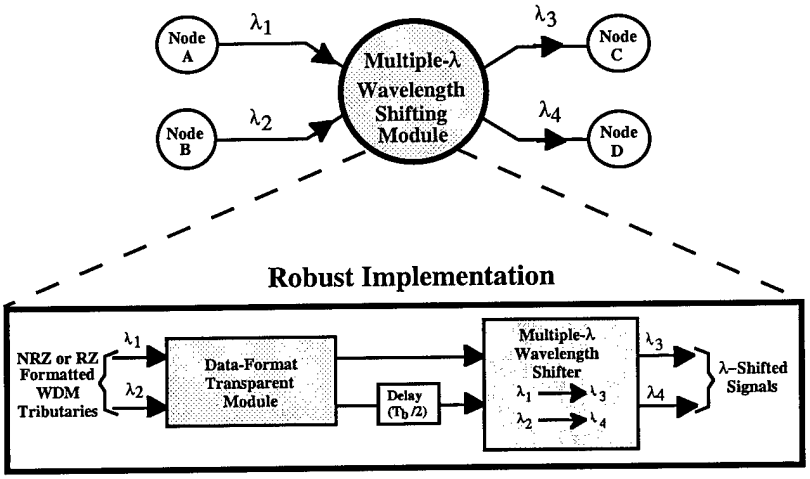

Fig. 3.7 A multiple-_ wavelength shifting module used to wavelength shift and route two independent WDM tributaries to two different wavelength destinations, and a robust conceptual implementation of such a module.

Figure 3.8 highlights the implementation of our multiple-_ wavelength shifting module. The experimental setup consists of an NRZ-to-RZ (NRZ→RZ) converter front-end followed by a multiple-_ wavelength shifter. We first discuss the operation of the data-format transparent front-end. This front-end consists of one semiconductor optical amplifier (SOA) whose injection current is directly modulated by the system clock. The clock signal has the effect of gating the SOA for half of every bit period. When the SOA gating is synchronized with the input data, each bit experiences a large gain during half of its bit period, while the SOA absorbs the input optical power during the other half of the bit period. This technique converts input NRZ signals into the RZ format as well as preserves input RZ formatted signals, thereby establishing NRZ and RZ data-format transparency for

our multiple-_ wavelength shifter. A critical requirement when using this NRZ→RZ conversion scheme is that the system clock must be recovered. One approach to deriving the required clock is to optically tap one of the input WDM channels, detecting the tapped power, and then recovering the clock with an electrical clock recovery circuit. In our demonstration, the required clock signal is provided by a local transmitter. The required synchronization of the SOA gating with the input data can be achieved by optically delaying the input signals while the clock is being recovered.

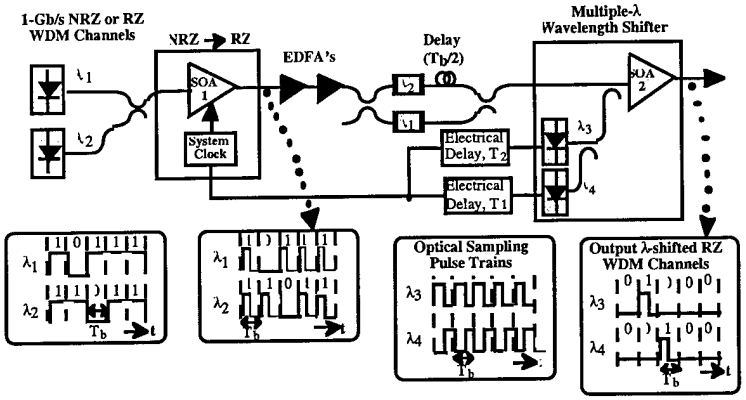

Fig. 3.8 The experimental setup for our format-transparent multiple-_wavelength shifting module. Input powers at SOA_1: -9.2 dBm → 1548-nm, -13.7 dBm → 1552-nm. SOA_1 gain-peak = 1568-nm, SOA_1 bias current = 40 mA. Input powers at SOA_2: -5.72 dBm → 1548-nm, -2.76 dBm → 1552-nm, -10.41 dBm → 1540-nm, -8.93 dBm → 1569-nm. SOA_2 gain-peak = 1550-nm, SOA_2 bias current = 180 mA

The conversion into the RZ format by the front-end is what enables the wavelength shifting of both input WDM channel tributaries from a single device. Since the SOA injection current is directly modulated in our experiment, the NRZ→RZ implementation described above is limited to bit rates up to ~5 Gb/s. A higher-speed NRZ→RZ implementation could be realized by using SOA cross-gain compression. In this case, the conversion is achieved by using an intense externally modulated clock-gated pump signal to saturate the SOA gain for half of every bit period. The clock-gated pump signal optically modulates the SOA gain and induces the conversion. This NRZ→RZ implementation is limited to bit rates of ~20 Gb/s, because the speed limitation is now due to the SOA gain recovery lifetime.

Once the input WDM channel tributaries (_1 and _2) are either preserved in the RZ format or converted to the RZ format, they are both amplified by two cascaded EDFAs and the WDM channel at _2 is *delayed* by half a bit period. This delay effectively interleaves the two RZ-converted WDM channels. This interleaving process is critical to the operation of the multiple-_wavelength shifter. The interleaved signals, at _1 and _2, and a pair of optical sampling pulse trains, at _3 and _4, are all coupled into an SOA cross-gain compression wavelength shifter (SOA_2). The optical sampling pulse trains alternately sample the input data from the two input RZ-converted WDM channels. The WDM channel at _1 is sampled only by the pulse train at _3, whereas the WDM channel at _2 is sampled only by the pulse train at _4. In other words, _1 acts as the pump for a _3 probe during the first half of the bit time, and _2 acts as the pump for a _4 probe during the second half of the bit time.

Within the multiple-_ wavelength shifter, the amplified and interleaved RZ-converted WDM channels invoke the cross-gain compression mechanism inherent within the homogeneously broadened SOA_2. This mechanism then simultaneously and independently encodes the *complement* of the data from each RZ-converted WDM channel onto the appropriate optical sampling pulse

train. This causes the WDM channel at _1 to be shifted to _3, whereas the WDM channel at _2 is shifted to _4. As a result of the NRZ→RZ conversion process as well as the interleaving of the input RZ-converted WDM channels, the data-format of the output wavelength shifted WDM channels is return-to-zero. Furthermore, the multiple-_ wavelength shifting module requires the input WDM channel tributaries to be synchronized.

Figure 3.9 shows the oscilloscope traces from our experimental demonstration. Two independent and synchronized NRZ 1-Gb/s WDM channels at 1548 and 1552 nm are simultaneously wavelength shifted to 1540 and 1569 nm, respectively. This demonstration represents both up-conversion (1552→1569 nm) and down-conversion (1548→1540 nm) over a 29 nm wavelength range; note the RZ format of the output wavelength shifted signals. The eye pattern for the up-conversion scenario (1552→1569 nm) is degraded compared to that for the down-conversion scenario because of the characteristic reduced extinction ratio when up-converting. The gain-peak of SOA_2 is 1550-nm and has a gain-bandwidth of ~25-nm.

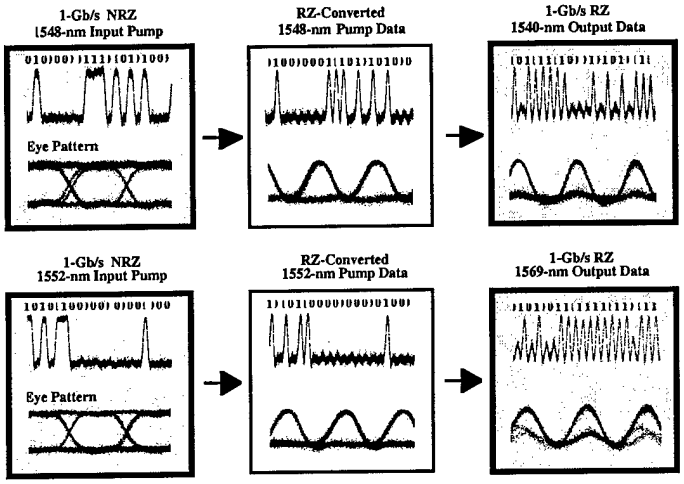

Fig. 3.9 Oscilloscope traces for our experimental demonstration showing simultaneous wavelength conversion over 29-nm of two independent WDM tributaries.

3.2.4 Multiple-Wavelength-Input All-Optical Wavelength-Shifting of Self-Routing Packets using Subcarrier-Multiplexed Control

Wavelength-division multiplexing (WDM) may enable highly functional and flexible optical networks in which wavelengths are used as routing paths. The use of high-speed all-optical wavelength (l) shifters may be critical for dynamically-reconfigurable networks and for networks requiring wavelength re-use due to an insufficient number of available wavelengths. One method of _-shifting uses semiconductor optical amplifier (SOA) cross-gain compression in which an intense optical pump signal modulates the SOA gain, inversely transferring the pump signal to a supplied weak CW probe on a different wavelength. Although the signal can be shifted to many output probe wavelengths, this method does not accommodate the shifting of a signal from more than one *input* wavelength at a time, thereby limiting network functionality. Moreover, no other wavelength-shifting method can accommodate such simultaneous wavelength shifting of multiple independent WDM input channels. Figure 3.10 shows a conceptual diagram of a wavelength-shifter which simultaneously and independently shifts each input wavelength to individual output wavelengths. Such a multiple-input-wavelength _-shifter would enable WDM network switching nodes to more efficiently route data packets located at several different possible wavelengths onto several possible free wavelengths which correspond to different destination nodes.

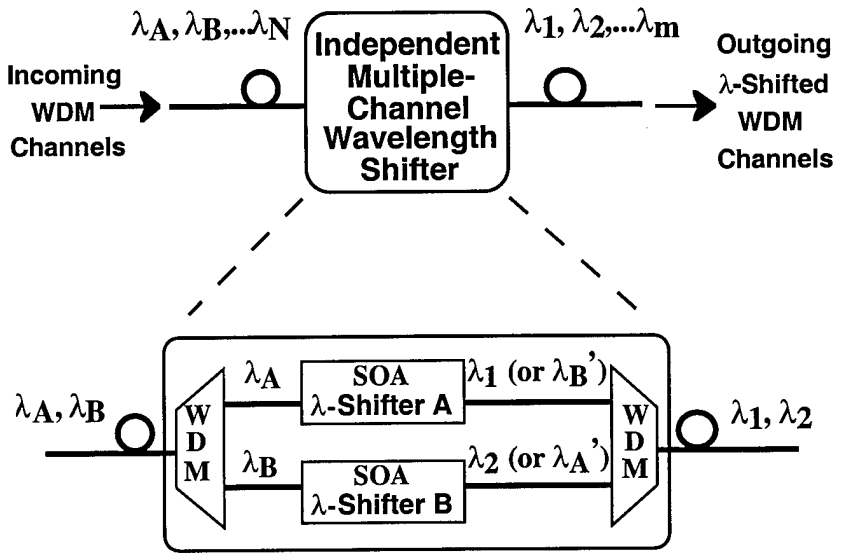

Fig. 3.10 Diagram of a multiple-channel wavelength shifter for a WDM routing node. The shifter is composed of spatially separation using a wavelength demultiplexer and parallel independent SOA-based wavelength shifters.

In this letter, we discuss a method for all-optical wavelength shifting of multiple-input-wavelengths in a WDM routing node. This technique involves spatial separation of incoming wavelengths followed by wavelength-shifting of the incoming signals using parallel independent SOAs (see Fig. 3.10). Subcarrier-multiplexed routing control as well as wavelength-interchange are incorporated to increase functionality. We measure near-error-free _-shifting of 1-Gb/s data for all cases. This technique for shifting multiple input wavelengths uses parallel SOAs to perform SOA cross-gain compression for each channel independently. However, this technique could also be accomplished by using parallel devices incorporating other forms of__-shiftingmethods, such as four-wave-mixing and Mach-Zehnder-based integrated modules.

The experimental setup is as follows. Packets are transmitted on two WDM input wavelengths _A (= 1541.9 nm) and _B (=1556.5 nm) using directly modulated DFB lasers. The transmitted signals are composed of 480-bit 1-Gb/s NRZ baseband data packets. This baseband data is multiplexed with a subcarrier frequency at either f_A = 1.4 GHz or f_B = 1.2 GHz which uniquely identifies the channel wavelength. The subcarriers are QPSK-modulated with 16-bit 50-Mb/s control headers which include flag bits. The combined transmitted WDM signal (baseband plus subcarrier) enters the multiple-channel _-shifterand is tapped and detected using a single 1.7-GHz subcarrier receiver. The headers on subcarriers f_A and f_B are recovered by using QPSK demodulators and the same oscillators as the transmitter subcarrier sources. The subcarrier headers are then used to instruct each electronic routing processor in a given parallel spatial path: (i) to perform flag detection and header processing, and, subsequently, (ii) to turn "ON" one of two possible probe lasers; note that one processor per input channel is employed. An important feature of the multiple-input _-shifter is that the input WDM signals are spatially separated using optical splitters and filters, and then wavelength shifted in parallel; an integrated frequency router could be used for easier wavelength separation.

One input WDM signal is coupled into SOA_A (gain peak = 1550 nm) and the other WDM signal is coupled into SOA_B (gain peak = 1565 nm). An EDFA is used to provide sufficient pump power for SOA cross-gain compression. The two possible probe signals for each SOA represent shifting of an individual WDM signal onto either: (i) a new available wavelength, or (ii) the other original input wavelength representing the function of wavelength interchanging. Based on the header information, packets on _A are either _-shifted to the other input channel's wavelength _B'= 1556.5 nm (i.e., wavelength interchanger function) or to an entirely different wavelength _1= 1534.6 nm.

Similarly, packets on _B are either _-shifted to the _A'= 1541.8 nm or to _2= 1571.1 nm. Note that primes (i.e., ') indicate wavelength-interchange. The pumps and the selected probe signals are coupled into the SOAs in a counter-propagating fashion in order to avoid having the pump signals appear at the output along with the probes. An angle-tuned 1-nm bandpass optical filter is used to select the appropriate probe wavelength to be passed to a 1.7-GHz baseband receiver at the output.

The use of subcarrier header routing control has the following advantages: (i) the data speed on the control subcarrier can be much lower than the data packet bit rate (i.e. <100 Mb/s), (ii) the RF subcarrier technology is relatively mature and relatively cost effective, and (iii) the header and baseband are sharing the same wavelength (as opposed to different wavelengths) and can co-propagate with other wavelengths in the same fiber without incurring dispersion-induced walk-off between packet and header or wasting valuable available wavelengths.

Figure 3.11 shows oscilloscope traces of _-shifted packets which are wavelength-shifted based on packet-header information. These oscilloscope traces are for six packet time slots, with A and B denoting the input wavelength; the arrows indicate the direction of wavelength shifting. Also shown are the SOA output spectra after EDFA post-amplification but without any optical filtering. Although pump-probe counter-propagation was used, the presence of the pump is still observed at the output due to small reflections. The optical powers P measured prior to entering the EDFA were: P_A= -3.5 dBm, P_B = -4.6 dBm, P_1 = -1 dBm, P_B' = -8 dBm, P_A' = -7 dBm, and P_2 = -22 dBm. Probe power P_1 was large in order to compensate for marginal coupling into one facet of SOA_A. In our experiment, both up-shifting and down-shifting of each input wavelength is performed. For the interchange case in which the output from both parallel paths are optically combined, optical filtering methods are needed to prevent the pump on one spatial path from interfering with the probe located at the same matching wavelength from the other parallel spatial path.

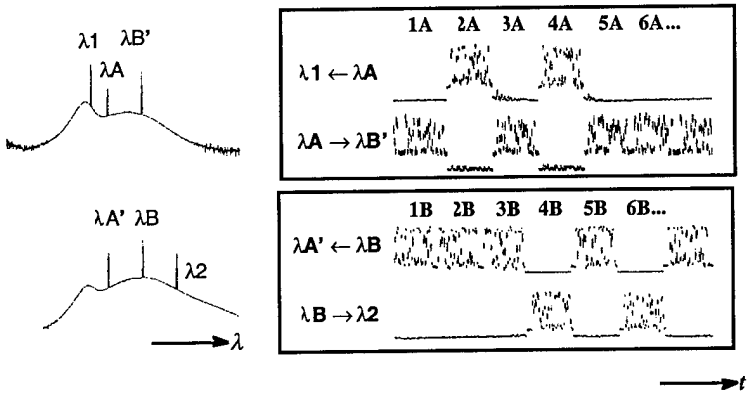

Fig. 3.11 Oscilloscope traces showing _-shifted self-routed packets. Left arrows indicate down-shifting, right arrows indicate up-shifting, and primes indicate wavelength interchange. Packets on input _A are either down-shifted to _1 or up-shifted to _B' using SOA_A. Packets on input _B are either down-shifted to _A' or up-shifted to _2 using SOA_B. Also shown are the post-amplified SOA output spectra without any optical filtering.

3.2.5 Experimental Demonstrations of All-Optical Conversions Between the RZ and NRZ Data Formats Incorporating Noninverting Wavelength Shifting Leading to Format Transparency

Future all-optical wavelength-division-multiplexed (WDM) networks may be required to support a variety of modulation and data formats. Two standard data formats that have found extensive use are the return-to-zero (RZ) and non-return-to-zero (NRZ) formats. Although the RZ data format requires twice the NRZ transmission bandwidth, it is quite useful in applications including passive time-division-multiplexing and -demultiplexing, soliton generation, and the suppression of stimulated Brillouin scattering. Also, since some optical processing operations unintentionally change the data format, fully functional WDM networks should have the capability of all-optically converting between the RZ and NRZ formats. Previous work has demonstrated all-optical RZ-to-NRZ conversion using a nonlinear optical loop mirror at 10-Gb/s, but the performance of this system was shown to be highly sensitive to the polarization state of the loop mirror.

We demonstrate a unique polarization insensitive semiconductor optical amplifier (SOA) based system that performs the desirable function of converting an RZ WDM channel into the NRZ format (RZ→NRZ). Our system allows the input RZ wavelength to be either preserved or wavelength shifted to a different wavelength during the conversion process. In an earlier paper, we demonstrated a complementary NRZ-to-RZ (NRZ→RZ) converter that uses SOA gain modulation to convert an input NRZ signal into an output RZ signal at the same wavelength. We use this NRZ→RZ converter and the RZ→NRZ converter presented in this work, to realize an all-optical NRZ→RZ→NRZ reconverter in which the original NRZ data format is recovered. The NRZ→RZ→NRZ operation may be necessary for the intermediate optical processing of packets. No previous work has demonstrated such a reconverter having this type of functionality. To demonstrate the robustness of this reconverter, we incorporate 4 EDFAs and 80 km of fiber between the individual NRZ→RZ and RZ→NRZ converters. The combination of the RZ→NRZ and NRZ→RZ→NRZ functions presented in this work may significantly help to realize RZ / NRZ data format transparency within dynamically reconfigurable and fully functional WDM networks. The fundamental operations used within our system to realize transparency in data format include optical sampling, wavelength shifting using cross-gain compression, time multiplexing and SOA gain modulation. We demonstrate our system by performing the RZ→NRZ and NRZ→RZ→NRZ conversion functions at 1-Gb/s while incurring low power penalties (<2 dB) for both cases.

Figure 3.12 (a) shows our system that implements the RZ→NRZ conversion while Fig. 3.12 (b) shows the conceptual operating mechanisms. The RZ→NRZ converter consists of two cascaded SOAs, two optical sampling pulse trains, and a CW probe signal. The physical mechanism behind this converter is SOA cross gain compression wavelength shifting. This mechanism relies on an intensity modulated pump signal and a low power CW probe signal which are simultaneously coupled into an SOA. The pump signal saturates the SOA gain only when its bit pattern is "HIGH". This cross gain compression results in an inverse modulation of the SOA gain that is available to the CW probe signal. The cross gain compression causes the complement of the pump data to be impressed onto the probe, effectively wavelength shifting the pump data from λ_{pump} to λ_{probe}.

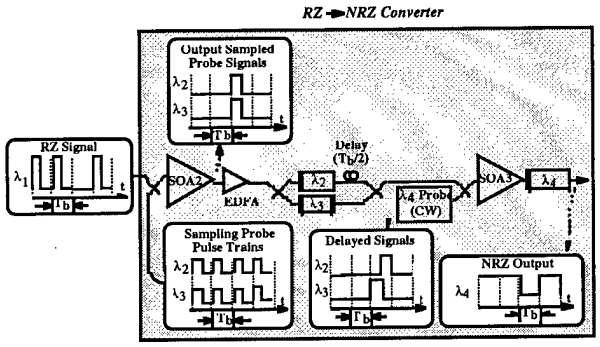

Fig. 3.12 (a) All-optical RZ→NRZ converter.

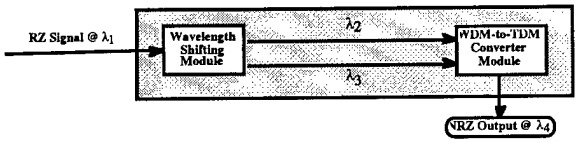

Fig. 3.12 (b) Conceptual implementation.

We exploit this principle to realize the RZ→NRZ conversion in the following manner. When an input RZ pump signal at $_1$ is coupled into SOA_1, it is optically sampled by a pair of synchronized low power RZ probe pulse trains at $_2$ and $_3$. These pulse trains can be generated by electronically recovering the input clock signal and then using this recovered clock to modulate a bank of two probe lasers. The required synchronization of the input RZ signal with the probe pulse trains for optical sampling can be accomplished by optically delaying the input RZ signal while the clock is being recovered.

To demonstrate the principle of this converter, in our demonstration the pulse trains are generated by modulating two probe lasers with the clock from a local transmitter, and the required synchronization is generated electronically. By optically sampling the input RZ pump signal, the complement of the pump data is impressed onto each probe pulse train by SOA_1 in a "broadcast" manner. At the output of SOA_1, both probe signals at $_2$ and $_3$ are amplified and filtered, after which the signal at $_2$ is delayed by half a bit. Both amplified and interleaved probe signals then become the pump signals for SOA_2, in which the complement of their data is wavelength shifted onto the input CW probe signal at $_4$ by the cross gain compression mechanism. This results in an NRZ signal at $_4$ having the same data polarity as that of the original RZ signal at $_1$. Hence, we have converted an RZ signal at l_1 into an NRZ signal at $_4$ with the data polarity preserved. Also, note that the wavelength of the output NRZ signal can be the same as that of the original RZ signal by setting $_4=_1$. Since this converter is based on cross gain compression, the operating bit rates are limited to ~20-Gb/s.

Figure 3.13 shows the 1-Gb/s oscilloscope traces from our demonstration in which: (a) an input 1571-nm RZ signal is sampled by two RZ probe pulse trains at 1552 and 1548 nm, and (b) the delayed and amplified probe signals (SOA_2 pumps) are multiplexed to create an output NRZ signal at 1540-nm. Figure 3.13 (c) shows the BER performance curves associated with this demonstration, in which a conversion power penalty of only ~0.8 dB at a 10^{-9} BER is incurred for a PRBS length of 2^{15}-1.

In order to realize complete RZ / NRZ format transparency, we demonstrate an NRZ→RZ→NRZ reconverter by cascading a NRZ→RZ converter with the RZ→NRZ converter. To show system robustness, the two converters are interconnected through a cascade of four EDFAs and 80 km of dispersion shifted fiber (DSF). The first three EDFAs are separated by 40 km of DSF, while the fourth EDFA immediately follows the third EDFA. Figure 3.14 shows the oscilloscope traces from our demonstration. In this case, a 1-Gb/s NRZ 1571-nm signal is first converted into the RZ format and is then wavelength shifted and reconverted back into a noninverted NRZ signal at 1540-nm. It is emphasized that if the cross gain compression mechanism is used to realize the NRZ→RZ conversion, the NRZ→RZ→NRZ reconverter is limited to operating bit rates of ~20-Gb/s.

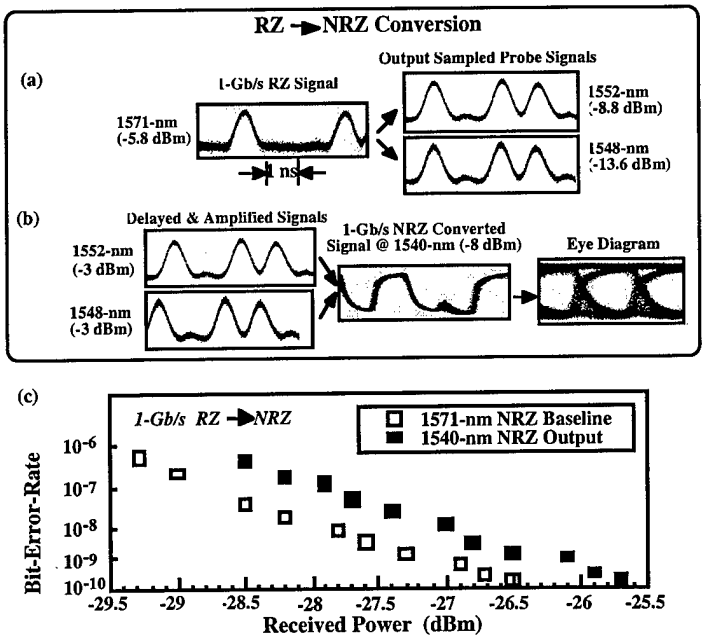

Fig. 3.13 (a) RZ→NRZ converter oscilloscope traces showing the input 1571-nm RZ signal and the output sampled probe signals. (b) RZ→NRZ converter oscilloscope traces showing the amplified and delayed pump signals and the converted NRZ output signal. (c) RZ→NRZ converter oscilloscope traces showing the bit-error-rate performance curves associated with this demonstration showing only a 1-dB power penalty at a 10^{-9} BER and for a PRBS length of 2^{15}-1.

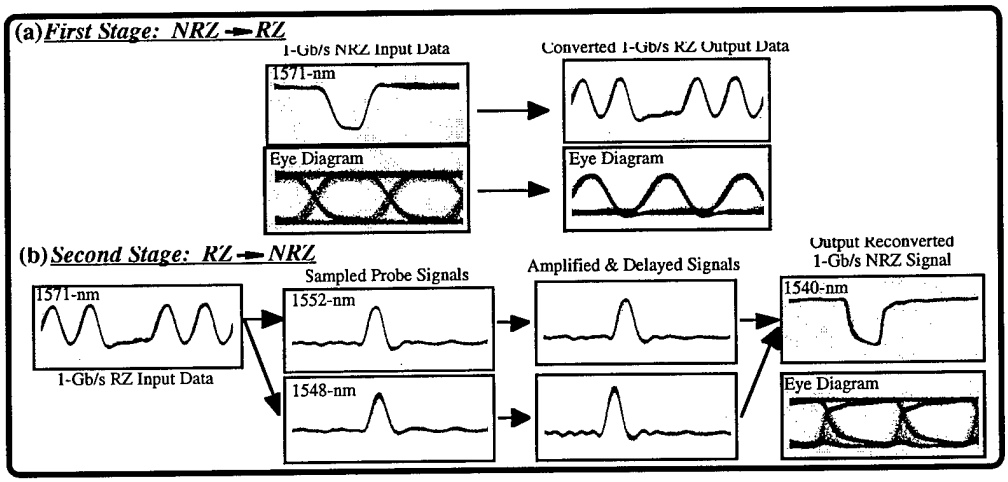

Fig. 3.14 (a) NRZ→RZ→NRZ oscilloscope traces showing the NRZ→RZ conversion process.

Fig. 3.14 (b) NRZ→RZ→NRZ oscilloscope traces showing the RZ→NRZ conversion process recovering the original NRZ formatted data.

3.2.6 A Polarization-Independent and Contrast-Ratio-Enhancing Module for All-Optical Wavelength Shifting Using SOAs

Wavelength shifting may play an important role in future wavelength-division-multiplexed (WDM) optical networks by allowing for dynamic routing and wavelength reuse. Furthermore, it is desirable that the optical signal remain in the optical domain throughout the transmission path in order to avoid optoelectronic bottlenecks. Several all-optical wavelength shifting schemes have been proposed, including cross-gain saturation in a semiconductor optical amplifier (SOA), saturable absorption in a DBR laser, and four-wave-mixing in an SOA. The mechanism of cross-gain saturation relies on an intense modulated pump signal on one wavelength causing significant inverse modulation of the SOA gain; when a cw probe signal is concurrently coupled to the SOA, the modulation of the input pump is then inversely transferred to the probe output. The cross-gain saturation method has a wide continuous conversion range (~ 40 nm), high conversion efficiency (> -10 dB) and can be implemented in a straight forward manner. However, the contrast ratio is significantly degraded upon upshifting, and the prospects for network cascadability with this method are not very promising. Recently, a two-stage configuration was demonstrated which alleviated the problem of poor contrast ratio performance. Another problem with cross-gain saturation is that the polarization dependence of the SOA gain: TE mode gain can be as much as ~5-6 dB larger than TM mode gain. This will cause the wavelength shifting performance to be dependent on the polarization of the incoming signal. In this letter, we demonstrate a polarization-independent and contrast-ratio-enhancing wavelength shifting module consisting of two polarization-dependent SOAs. We perform 1 Gb/s wavelength upshifting over 19 nm and reduce the power penalty for the shifter module from 5 dB to 1.5 dB and the polarization dependence from 3.5 dB to 0.5 dB in comparison to a single-SOA-based wavelength shifter.

Figure 3.15 illustrates the conceptual diagram of the polarization-insensitive and contrast-ratio-enhancing module. The modulated pump is split into two branches, $pump_1$ and $pump_2$ feeding into SOA_1 and SOA_2, respectively. The CW probe is coupled into the SOAs from the opposite direction. The output probe from SOA_1 is inversely modulated by $pump_1$, obtaining a contrast ratio of CR_1 (dB). When this probe signal enters SOA_2, it is synchronized with the $pump_2$ by a delay line. $Pump_2$ then modulates the gain to the probe, inducing an additional contrast ratio (CR_2) to the probe. The resultant total contrast ratio CR equals CR_1+CR_2. This phenomenon was first demonstrated in reference by using specially fabricated polarization independent SOAs. Instead, we use standard polarization-dependent SOAs. We introduce a polarization controller (PC) for $pump_2$ so that the polarization of the pump to SOA_1 and SOA_2 are always orthogonal ($SOA_1 \perp SOA_2$). That is, if the $pump_1$ input to SOA_1 is TE, then $pump_2$ input to SOA_2 is TM, and vice versa. The net contrast ratio is always: CR = CR(TE) + CR(TM). This is similar to the method of constructing a polarization insensitive SOA by cascading two orthogonally polarized SOAs. Consequently, our shifting module is not only contrast-ratio-enhancing but also polarization-insensitive.

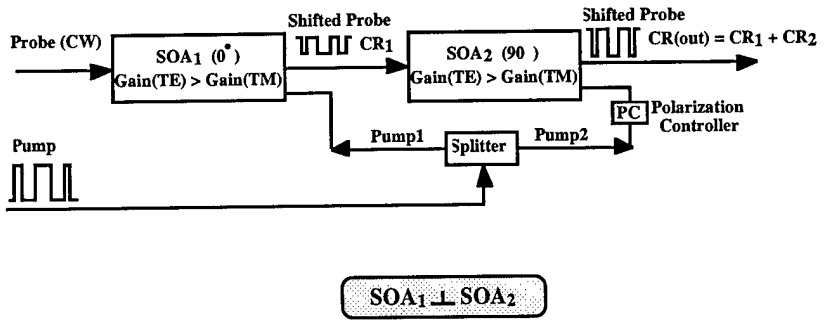

Fig. 3.15 Schematic of the polarization-independent and contrast-ratio-enhancing module.

A diagram of the experimental setup is shown in Fig. 3.16. The pump light at 1552 nm is directly modulated by a 2^{15}-1 PRBS from the BER transmitter. SOA_1 is biased at 180 mA with a peak wavelength at 1567 nm. SOA_2 is also biased at 180 mA with a peak wavelength at 1550 nm. The insertion loss for the pump to SOA_1 and SOA_2 from the input port of the first 3 dB coupler is 7 dB and 9.5 dB respectively. Insertion losses from the PC and delay line contribute an additional 2.5 dB for SOA_2. The pump power is measured before the input port of the first coupler. The wavelength shifted signal out of SOA_2 is filtered by a 1 nm filter and coupled into an optical receiver, amplified and input to the BERT receiver.

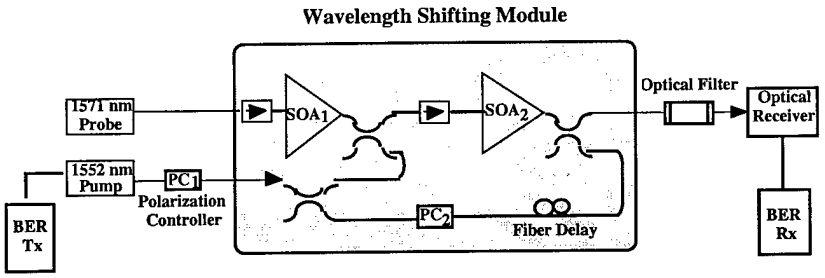

Fig. 3.16 Experiment setup of the robust module.

Figures 3.17 (a) and 3.17 (b) show the polarization dependence of the contrast ratio for the output shifted probe as a function of the input pump contrast ratio for a single SOA and for our two-SOA module. Figure 3.17 (a) shows that both SOA_1 and SOA_2 are strongly polarization dependent. The polarization state of pump is adjusted by rotating PC_1 by 360°, thereby allowing us to find out the best contrast ratio (TE mode) and the worst contrast ratio (TM mode) for all possible polarizations. For an input contrast ratio of 10 dB, the polarization dependence is 2.2 dB and 2.4 dB for SOA_1 and SOA_2 respectively. When the contrast ratio of an input pump at TM mode is 10 dB, the contrast ratio of the wavelength shifted probe only yields 3.2 dB and 2 dB for SOA_1 and SOA_2,, respectively. Because the system performance is measured by the worst case scenario, the polarization dependence of the SOA will pose a serious problem due to severe contrast ratio degradation (from 10 dB down to 2 dB) for the TM mode input pump signal conversion. Figure 3.17 (b) illustrates the polarization dependence of wavelength shifting for different polarization alignment of the two-SOA-based module. In the case of the polarizations of SOA_1 and SOA_2 being parallel (SOA_1 // SOA_2), the polarization state of pump is again adjusted by rotating PC_1 by 360°, thereby allowing us to find out the best and worst contrast ratio for all possible polarizations. The polarization dependence for 10 dB input contrast ratio is more than 4 dB. This increased polarization dependence is caused by the accumulating of the polarization dependencies of the two individual SOAs. In the case of (SOA_1 ⊥ SOA_2), we observe only a 1.1 dB polarization dependence for a 360° polarization rotation of the input pump, which is much smaller than the polarization dependence of an individual SOA. Secondly, the worst contrast ratio still yields ~6.5 dB for an input contrast ratio 10 dB, which is still larger than the contrast ratio of the TE input for a single SOA. The residual 1.1 dB polarization dependence is attributed to the two SOAs not being perfect matched.

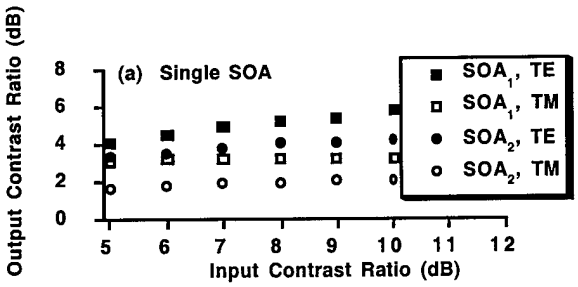

Fig. 3.17 (a) The contrast ratio of wavelength shifted signal as a function of the contrast ratio of input pump for single SOA. The pump power at a "1" is 6 dBm. The probe power is -5 dBm and -10 dBm for SOA_1 and SOA_2 respectively.

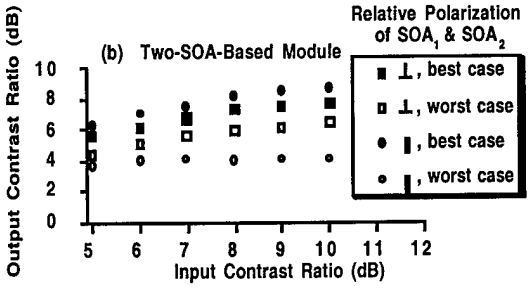

Fig. 3.17 (b) The contrast ratio of the wavelength shifted signal as a function of contrast ratio of the input pump for the two-SOA-based module. The pump power at a "1" is 6 dBm and the probe power is -5 dBm.

Figure 3.18 (a) shows the polarization dependence of the bit-error-ratio (BER) measurements. The polarization dependence is defined as the differential of sensitivities at BER of 10^{-9} for all possible polarizations of input pump signal. Polarization dependence of ~3 dB and ~4 dB is measured for SOA_1 and SOA_2, respectively. As expected, the TM mode pump incurs a big power penalty (~5 dB) due to contrast ratio degradation, which will severely limit the system performance. Figure 3.18 (b) shows the polarization dependence of our proposed module (SOA1 ⊥ SOA2). The polarization dependence decreases to merely 0.5 dB and the largest power penalty reduces to only 1.5 dB. This significant improvement is due to the reduced polarization fluctuation of the contrast ratio over much enhanced contrast ratio. Finally, the proper polarization alignment is critical to achieve this improvement. In the case of (SOA1 // SOA2), the polarization dependence increases to 2.5 dB and the largest penalty increases to 3 dB.

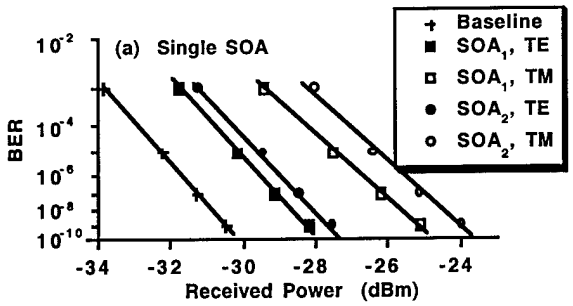

Fig. 3.18 (a) The BER measurements of the wavelength shifted signal for a single-SOA

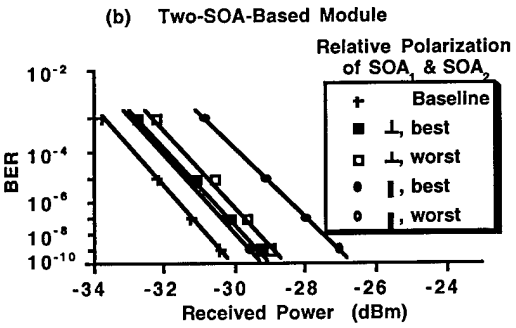

Fig. 3.18 (b) The BER measurements of the wavelength shifted signal for two-SOA-based wavelength shifting module. The pump power is 3 dBm and the probe power parameters are the same as those in figure 3.17.

3.3 Bibliography

Refereed Journals

1. W. Shieh, E. Park, and A.E. Willner, "Demonstration of Output-Port Contention Resolution in a WDM Switching Node Based on All-Optical Wavelength Shifting and Subcarrier-Multiplexed Routing Control Headers," **IEEE Photonics Technology Letters**, vol. **9**, pp. 1023-1025, 1997.
2. W. Shieh and A.E. Willner, "A Wavelength-Routing Node Using Multifunctional Semiconductor Optical Amplifiers and Multiple-Pilot-Tone-Coded Subcarrier Control Headers," **IEEE Photonics Technology Letters**, vol. **9**, pp. 1268-1270, 1997.
3. D. Norte and A.E. Willner, "Experimental Demonstration of a Multiple-Wavelength Wavelength Shifter for Dynamically Reconfigurable WDM Networks," **IEEE Photonics Technology Letters,** vol. **9**, pp. 922-924, 1997.
4. E. Park and A.E. Willner, "Network Demonstration of Self-Routing Wavelength Packets Using an All-Optical Wavelength Shifter and QPSK Subcarrier Routing Control," **IEEE Photonics Technology Letters**, vol. **8**, pp. 938-940, 1996.
5. D. Norte and A.E. Willner, "Demonstration of an All-Optical Data Format Transparent WDM-to-TDM Network Node With Extinction Ratio Enhancement for Reconfigurable WDM Networks," **IEEE Photonics Technology Letters**, vol. **8**, pp. 715-717, 1996.
6. W. Shieh, E. Park and A.E. Willner, "All-Optical Wavelength Shifting of Microwave Subcarriers by Using Four-Wave Mixing in a Semiconductor Optical Amplifier," **IEEE Photonics Technology Letters,** vol. **8**, pp. 524-546, 1996.
7. D. Norte and A.E. Willner, "Experimental Demonstrations of All-Optical Conversions Between the RZ and NRZ Data Formats Incorporating Noninverting Wavelength Shifting Leading to Format Transparency," **IEEE Photonics Technology Letters,** vol. **8**, pp. 712-714, 1996.
8. W. Shieh, S.H. Huang, and A.E. Willner, "A Polarization-Independent and Contrast-Ratio-Enhancing Module for All-Optical Wavelength Shifting Using SOA's," **IEEE Photonics Technology Letters,** vol. **8**, pp. 533-535, 1996.
9. D. Norte and A.E. Willner, "All-Optical Data Format Conversions and Reconversions Between the Wavelength and Time Domains for Dynamically Reconfigurable WDM Networks," **IEEE/OSA J. of Lightwave Technology and IEEE J. on Selected Areas in Communications**, Special Issue on Multiple-Wavelength Technologies and Networks, vol. **14**, pp. 1170-1182, 1996.
10. A.D. Norte and A.E. Willner, "Multi-Stage All-Optical WDM-to-TDM-to-WDM and TDM-to-WDM-to-TDM Data Format Conversion and Reconversion Through 80 km of Fiber and 3 EDFAs," **IEEE Photonics Technology Letters,** vol. **7**, pp. 1354-1356, 1995.
11. W. Shieh and A.E. Willner, "Optimal Conditions for High-Speed All-Optical SOA-Based Wavelength Shifting," **IEEE Photonics Technology Letters,** vol. **7**, pp. 1273-1275, 1995.

Invited Papers

1. A.E. Willner, "Wavelength-Division-Multiplexing for Highly-Functional Optical Networking," **Invited Paper, IEEE Network Magazine**, early 1998.
2. A.E. Willner, "Dynamic Multiple-Wavelength Systems and Networks," **Invited Paper, Society of Photo-Instrumentation Engineers (SPIE) Conference on Optics,** Communications Symposium, Taipei, Taiwan, July, 1998.
3. Alan E. Willner, Bogdan Hoanca, and Timothy Day, "Dynamically Reconfigurable WDM Networks Remain a Challenging Goal," **Invited Paper, Lightwave Magazine**, June, pp. 54-57, Pennwell Publishers, 1997.

4. A.E. Willner, "Dynamically-Reconfigurable WDM Networks," **Invited Paper, Optoelectronics and Communications Conference (OECC) '97**, Seoul, Korea, July 1997.
5. A.E. Willner, "Systems Requirements of WDM Components," **Invited Paper, IEEE LEOS Summer Topical Meeting on WDM Components**, Proceedings, August 1997, Montreal, Canada (IEEE/LEOS, Piscataway, NJ, 1997).
6. A.E. Willner, "Advanced Multimedia Communication Systems," **Invited Paper, World Conference on The Role of Advanced Materials in Sustainable Development, Chemrawn IX**, International Union of Pure and Applied Chemistry, Seoul, Korea, Sept., 1996.
7. A.E. Willner, "Reconfigurable WDM Networks," **Invited Paper, Conf. on Lasers and Electro-Optics (CLEO)**, paper CFG7, Anaheim, CA, June 1996 (OSA, Wash., D.C., 1996).
8. A.E. Willner, E. Park, D. Norte, and W. Shieh, "Systems Applications of All-Optical Wavelength Shifting," **Invited Paper, Meeting of the Optical Society of America**, Rochester, NY, Oct. 1996 (Optical Society of America, Wash., D.C., 1995).
9. A.E. Willner, "Applications of All-Optical Wavelength Shifting to Reconfigurable WDM Networks," **Society of Photo-Instrumentation Engineers (SPIE) Photonics East Conference on Emerging Components and Technologies for All-Optical Photonic Systems,** Technical Digest, Nov., 1996, Boston, MA (SPIE, Bellingham, Wash., 1996).
10. A.E. Willner, "Overview of Systems Issues for WDM Components," **Invited Paper, Society of Photo-Instrumentation Engineers (SPIE) Photonics West Conference on Wavelength Division Multiplexing Components**, Technical Digest, paper 2690-03, Jan., 1996, San Jose, CA (SPIE, Bellingham, Wash., 1996).

Refereed Proceedings

1. S. Schröder, R. Teschendorf, and A. E. Willner, "Contention Resolution of High-Speed WDM Packets Using a Dynamically- Controlled Multiple-Wavelength Fiber Loop Buffer and Wavelength Shifting," **Conference on Lasers and Electro-Optics**, Baltimore, MD, May 1997 (Optical Society of America, Wash., D.C., 1997).
2. T. Sangsiri, M. I. Hayee, B. Hoanca, W. Shieh, and A. E. Willner, "Stability and Dynamic Range of a Mach-Zehnder Wavelength Shifter," **Conference on Optical Fiber Communications** '97, paper WL52, Dallas, TX, Feb. 1997 (Optical Society of America, Washington D.C., 1997).
3. M. I. Hayee and A. E. Willner, "Pre-Compensation of Dispersion and Nonlinearities in 10 and 20 Gb/s WDM Systems," **Conference on Optical Fiber Communications** '97, paper WL39, Dallas, TX, Feb. 1997 (Optical Society of America, Washington D.C., 1997).
4. W. Shieh and A.E. Willner, "Demonstration of Output-Port Contention Resolution in a 2X2 WDM Switching Node Based on All-Optical Wavelength Shifting and Subcarrier-Multiplexed Routing-Control Headers," **Conference on Optical Fiber Communications** '96, **Post-Deadline paper** PD-36, San Jose, CA, Feb. 1996 (Optical Society of America, Washington D.C., 1996).
5. E. Park, D. Norte, and A.E. Willner, "Demonstration of Multiple-Wavelength-Input All-Optical Wavelength-Shifting Spatial and Temporal Techniques with Subcarrier-Multiplexed Control for Self-Routing," **Conference on Optical Fiber Communications** '96, **Post-Deadline paper** PD-34, San Jose, CA, Feb. 1996 (Optical Society of America, Washington D.C., 1996).
6. W. Shieh and A.E. Willner, "A Wavelength-Routing Node by Using Multifunctional Semiconductor Optical Amplifiers and Multiple-Pilot-Tone-Coded Subcarrier Control Headers," IEEE LEOS Summer Topical Meeting on **Broadband Optical Networks**, paper MA5, Keystone, Colorado, Aug. (IEEE, Piscataway, New Jersey, 1996).

7. D. Norte and A.E. Willner, "Demonstrations of All-Optical Conversions Between the RZ and NRZ Data Formats Incorporating Noninverting Wavelength Shifting Leading to Format Transparency," IEEE LEOS Summer Topical Meeting on **Broadband Optical Networks**, paper MB5, Keystone, Colorado, Aug. (IEEE, Piscataway, New Jersey, 1996).

8. E. Park and A.E. Willner, "Survivability of QPSK-Encoded Subcarrier Signals in an All-Optical Wavelength-Shifting System," **Conference on Lasers and Electro-Optics**, paper CFG7, Anaheim, CA, June 1996 (Optical Society of America, Wash., D.C., 1996).

9. W. Shieh and A.E. Willner, "SNR Improvement of Four-Wave Mixing Wavelength Shifting by Noise Prefiltering in a Semiconductor Optical Amplifier," **Conference on Lasers and Electro-Optics**, paper CThB5, Anaheim, CA, June 1996 (Optical Society of America, Wash., D.C., 1996).

10. E. Park and A.E. Willner, "Network Demonstration of Self-Routing Wavelength Packets Using an All-Optical Wavelength Shifter and QPSK Subcarrier Routing Control," **Conference on Optical Fiber Communications** '96, paper WD6, San Jose, Feb. 1996 (OSA, Washington D.C., 1996).

11. W. Shieh, S.H. Huang, and A.E. Willner, "A Polarization-Independent and Contrast-Ratio-Enhancing Module for All-Optical Wavelength Shifting Using SOA's," **Conference on Optical Fiber Communications** '96, paper WG5, San Jose, CA, Feb. 1996 (Optical Society of America, Washington D.C., 1996).

12. W. Shieh, E. Park and A.E. Willner, "All-Optical Wavelength Shifting of Microwave Subcarriers by Using Four-Wave Mixing in a Semiconductor Optical Amplifier," **Conference on Optical Fiber Communications** '96, paper WH4, San Jose, CA, Feb. 1996 (OSA, D.C., 1996).

13. D. Norte and A.E. Willner, "Demonstration of an All-Optical Data Format Transparent WDM-to-TDM Network Node With Extinction Ratio Enhancement for Reconfigurable WDM Networks," **Conference on Optical Fiber Communications** '96, paper WD5, San Jose, CA, Feb. 1996 (Optical Society of America, Washington D.C., 1996).

14. D. Norte and A.E. Willner, "Simultaneous Probe and Pump Extinction Ratio Enhancement Demonstration in All-Optical Noninverted Wavelength Shifting," **Conference on Optical Fiber Communications** '96, paper WM7, San Jose, CA, Feb. 1996 (Optical Society of America, Washington D.C., 1996).

4.0 Pruned Octree Feature for Interactive Retrieval, C.-C. Jay Kuo

Low-level features such as the color, texture and shape of objects have been widely studied for similarity search in image indexing and retrieval. A new color indexing scheme based on the octree quantization scheme is proposed in this research to achieve efficient multiresolution image retrieval. The new color feature not only integrates commonly used color features such as the color histogram and dominant color, but also support a selective filtering strategy to speed up the retrieval process. It can also be further combined with other visual features to facilitate similarity searching. Extensive experiments are performed to illustrate the performance of the proposed approach.

4.1 Introduction

Advances in modern technologies have led to huge and ever growing archives of sounds, images, and videos, in diverse application areas such as medicine, remote sensing, industry, engineering, entertainment, education and on-line information services. This is similar to the situation that occurred during the earlier development of computer technologies, in which the rapidly increasing amount of alpha-numeric data resulted in the database management system (DBMS). A DBMS is designed to organize a large amount of data into structured records, which are indexed by key attributes so that information retrieval and storage are convenient and efficient. However, this system does not work well for multimedia information management because of the difficulties in several aspects: the diversity of the data (e.g. image, video, audio), the large capacity of the unit record (e.g., a raw gray level image with size 512 by 512 has 256 kb before compression), and lack of semantic meaning of the data at the physical level (e.g. no semantic meaning at the pixel level for images). To exploit the full benefit of the explosive growth of information, there is a strong demand in the development of efficient techniques for the storage, browsing, indexing, and retrieval of multimedia data [1, 2].

Effective retrieval of image data is an important building block for multimedia information management. For an image to be searchable, it has to be indexed by its content which is either annotated by manually entered keywords or described by automatic extracted features. Although it seems effortless for a human being to recognize a friend's face in a picture, or to find out photos of horses from a collection of pictures of animals, object recognition and classification are still among the most difficult problems in image understanding and computer vision. In a small image database, it is easier to manually annotate a picture of horses by the keyword "horse" than to use the computer to recognize a horse with a training program, which may need to analyze various visual features such as shapes, colors, object occlusions and view points, etc. Unfortunately, manual annotation of large image databases can involve with a prohibitive amount of labor. In addition, a limited number of keywords are usually not sufficient to describe the details in a content abundant image. In order to gain access to images based on their contents, low-level features such as colors [3,4], textures [5,6], and the shapes of objects [7] are widely used as indexing features for image retrieval to circumvent the difficulties of image understanding.

Among the low level features, color information has been extensively studied because of its robustness with respect to scaling, orientation, perspective and occlusion of images. Color features that were intensively used in image retrieval include global and local color histograms, the mean (i.e. average color) and the higher order moments of the histogram [8]. The average and dominant colors can help filtering out irrelevant images without too much computational cost, but they do not support a detail comparison of color appearance among images. The global color histogram provides a good approach to the retrieval of images that are similar in overall color contents. There has been research work to improve the performance of color-based extraction methods. For example, the QBIC (Query by Image Content) [1] system supports color feature extraction of manually outlined objects. The evaluation study made by Zhang and Smoliar [4] showed that the fixed size local histogram is computationally simple and efficient in some applications. The color indexing method proposed by Stricker and Dimai [12] extracted color features defined in fuzzy regions adaptive to image content.

There are common issues underlying all color-based retrieval methods. They are the selection of proper color spaces in which image colors are represented [9], the use of proper color quantization methods to reduce the color resolution, and the development of efficient feature representations to support an efficient indexing method and a flexible query process. We have investigated the effect of

color quantization on the performance of image retrieval. The results were reported in [9, 10]. We observed that the fixed color quantization scheme used in the extraction of features such as the global and local histogram has one major drawback. That is, similar colors might be quantized to different buckets in the histogram, thus leading to false misses. In this work, we will investigate a new color feature based on multiresolution color clustering. The new color feature is more efficient than the multiresolution color histogram described in our previous work in the sense that it can provide the color feature of images according to the naturally color clustering rather than the fixed bucket quantization. We also developed a set of filtering methods based on the new color feature to facilitate the retrieval process, including filtering by the dominant color, by the color depth, and by tree intersection. A combination of these methods provides a prompt access of images in the data base.

4.2 Similarity Measurement of Images

Similarity measurement of images can be classified into three levels: pixel matching level, feature matching level, and semantic meaning level. Pixel level similarity comparison using L-1 and L-2 distance is straightforward but lack of robustness to image scaling, rotation and translation thus is seldom used. Similarity comparison by semantic meaning matching is ideal for retrieval but is lack of technique support of image understanding. Similarity comparison by extracted features are widely used currently. One typical problem in feature-based image access is "query by example", i.e. to search images in the database which are similar to a given query image. However, the meaning of similarity is quite vague. It might refer to the similarity in color appearance of pictures, in the texture of objects, or in the facial expression of people, etc. An interactive query process should be applied to refine the query so that the "similarity" defined by a specific user for a particular situation can be approached gradually. In this work, we demonstrate a possible solution of interactive refinement of "similarity" using pruned octree color feature. It solves a special class of similarity matching problem, i.e. searching images similar in color appearance.

4.2.1 Single Resolution Measurement

The color histogram of an image describes its color distribution. Every pixel in the image corresponds to a point in a 3-D color space. A similar image set can be selected based on the color distribution

$$\{T \mid dist(H_Q, H_T) < \varepsilon\},$$

where H_Q and H_T are color histograms of the query and target images, respectively, at the finest resolution level. That is, if a pixel is described by R, G and B color components of R-bit each, then H_Q and H_T are defined on the cubic lattice of $2^R \times 2^R \times 2^R$ points. However, the resolution of H_Q and H_T are too high to be used in practice. To simplify the computation, the color space has to be quantized to reduce the color resolution. Thus, histograms defined on the quantized space are used to get the similarity set, and we can get a new similar image set based on

$$\{T \mid dist(\hat{H}_Q, \hat{H}_T) < \varepsilon\},$$

where $\hat{H}_Q$ and $\hat{H}_T$ are quantized histograms of the query and target images, respectively. The quantization schemes in obtaining $\hat{H}_Q$ and $\hat{H}_T$ should be the same, i.e. one color quantization scheme is used for all images within the database.

A quantized histogram is usually represented by an N-dimensional vector, where N is the total number of quantization bins. For example, a RGB color histogram, which has been quantized into k bins for R, l bins for G, and m bins for B, can be represented as a vector of dimension $N = k \times l \times m$. Different similarity metrics for histograms have been studied. One example is the histogram intersection technique proposed by Swain and Ballard [3] defined as

$$\frac{\sum_{i=1}^{N} \min\left(\hat{H}_Q(i), \hat{H}_T(i)\right)}{\sum_{i=1}^{N} \hat{H}_Q(i)}$$

Another similarity metric [11], which takes into account the perceptual similarity between bins of histograms, was proposed by Hafner, Sawhney. It is defined as

$$dist(\hat{H}_Q, \hat{H}_T) = (\hat{H}_Q - \hat{H}_T)^T A(\hat{H}_Q - \hat{H}_T) = \sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij}(\hat{H}_Q(i) - \hat{H}_T(i))(\hat{H}_Q(j) - \hat{H}_T(j))$$

where matrix $\mathbf{A} = [a_{ij}]$ contains similarity weighting coefficients between colors corresponding to bins i and j.

4.2.2 Multiresolution Measurement

There are several drawbacks in the single resolution quantization method. First, as an indexing feature, the discriminating ability of the color histogram is determined by the selection of the quantization method (i.e. color resolution). The computational complexity increases quickly as the resolution of color feature increases. This can be a major problem in applications where the desired performance requires a high resolution of color features. Second, the histogram obtained by a single resolution quantization is not efficient in the sense that many buckets are empty, since it is often that colors in a given image only occupy a small subspace of the entire color space. Third, it is observed that results of quantized images are very sensitive to the location of quantization boundaries. As shown in Figure 4.3.2.1, two similar colors can be quantized into two totally different bins. As a result, image B can not be included in the similarity set of image A.

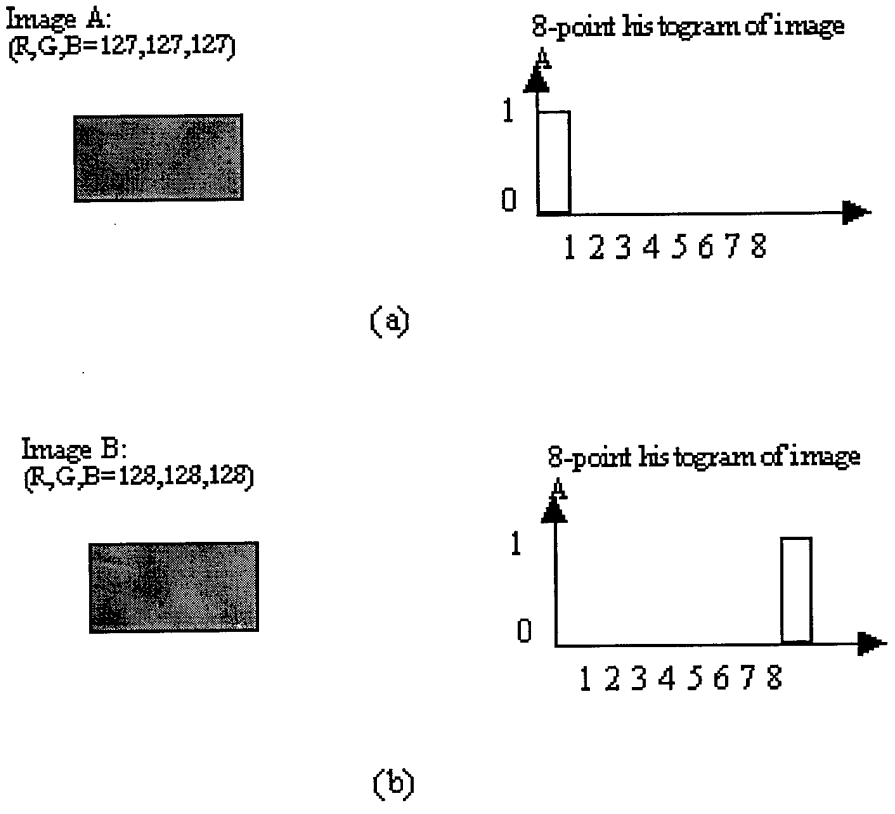

Figure 4.1 The problem of histogram mismatch with the single resolution quantization method.

To overcome the disadvantages of the single resolution quantization method, it is desirable to develop a set of multiresolution criteria to compare the color distance among images:

$$dist\left(H_Q, H_T\right)^{(i)} = dist\left(f_Q^{(i)}, f_T^{(i)}\right),$$

where $f_Q^{(i)}$ and $f_T^{(i)}$ represent color features of the query and target images at the ith resolution, respectively. The lowest resolution ($i=1$) feature is used to compare the similarity of images on the entire database and get a candidate-image set with a very low computational complexity. Comparison of higher resolution features is then performed within the candidate-images set to reduce the computational cost. To avoid the problem of putting similar colors into different buckets, one possible solution is to use a color clustering technique rather than the fixed quantization boundary in obtaining the color features $f_Q^{(i)}$ and $f_T^{(i)}$ at the ith level. The construction of a new multiresolution color feature by color clustering and the corresponding similarity measurement will be described in detail in the next section.

4.3 Multiresolution Color Representation with Pruned Octree

One simple approach to extract multiresolution color features is based on the octree color quantization, which will be briefly reviewed in Section 4.3.1. Another way to get the lowest resolution feature $f_T^{(i)}$ is to split the color space into 8 subspaces and calculate the average color and the number of pixels within each subspace. Recursively splitting each subspace will lead to color features $f_T^{(i)}$ in different resolutions. In many cases, the natural color cluster might lie across the boundaries of the simple octree quantization method. When similar colors are quantized to into different bins, we have false misses. To overcome this problem, we propose to cluster similar colors by merging octree nodes to represent the natural color clustering of each individual image. This process is described in Section 4.3.2.

3.1 Octree Initialization

Figure 4.2 shows the structure of an octree and its relationship with the RGB color space. As shown in Figure 4.2 , at the first level of the tree, the 8 children of the root corresponds to the eight subspaces of the entire space. Similarly, each of the eight nodes can have its own 8 children corresponding to further divided subspaces. The maximum depth of the octree for representing 24-bit image is 8; each leaf corresponds to one of the 65536 colors. Each color defines a unique path through the octree from the root to the leaf nodes. Two quantities can be used to describe the color information within a subspace: the number of pixels within it and the average color of the pixels. Thus, each node has two attributes, i.e. the normalized number N of pixels and the average color $\vec{C}$ of pixels. $\vec{C}$ together with N provide a better description of color distribution. When a color is inserted to the octree, its path is traced. The attributes of the intermediate nodes are modified while insertion. Note that similar colors will share a common path to some intermediate node so that color quantization can be done by mapping similar colors to the color of an intermediate node.

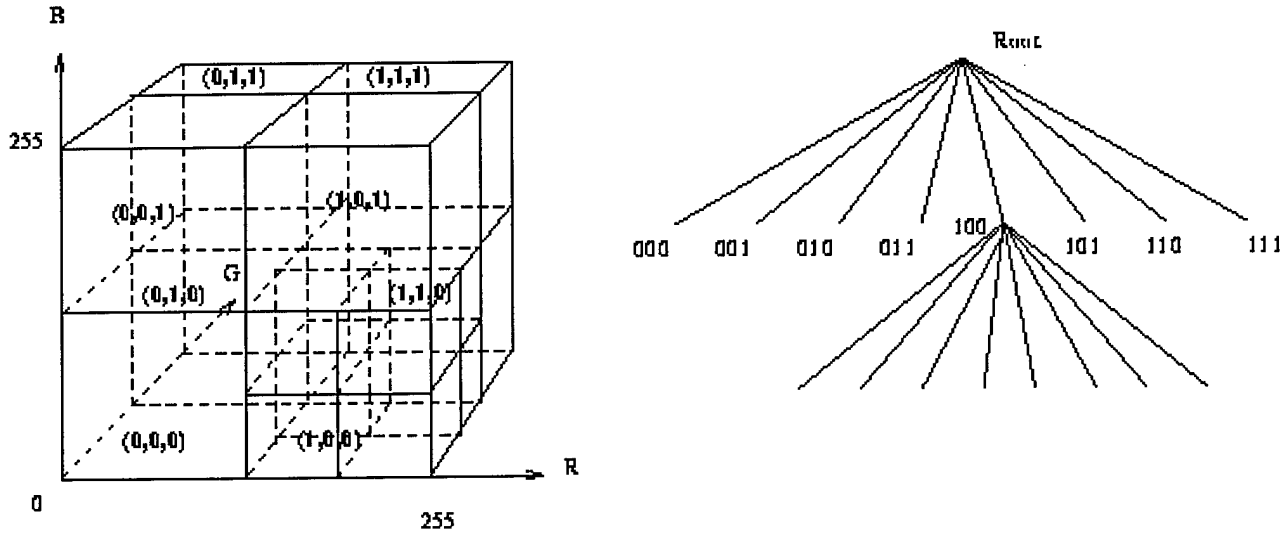

Figure 4.2 Octree structure.

The octree of an image can be obtained by scanning the color value of each pixel and inserting it into the octree. The easiest way to explain the insertion process is to consider an example shown in Figure 4.3, where the procedure of inserting a pixel with color components R=53 (00110101 in binary), G =187 (1011101 in binary) and B=197 (11001111 in binary) into a 8-level octree is illustrated. It is worthwhile to point out that it is usually unnecessary to use all 8 levels of the octree for color quantization. The depth of the octree of all images in our experimental database is in fact less than or equal to 4 after tree reduction. Consequently, it is also not necessary to reach the full depth of the octree in the insertion process. In the example of Figure 4.3, the nodes being traced until the 4th level are (0,1,1), (0,0,1), (1,1,0) and (1,1,0), which are the combination of the first, second, ..., fourth bits of the tree primary color.

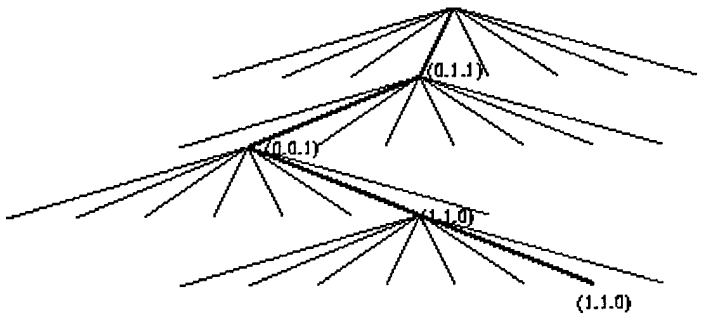

Figure 4.3 An example of inserting a color point with R=53, G=187, B=197.

4.3.2 Octree Pruning

A complete octree contains too many fine details in representing the color feature of an image. For example, a 4-level octree has $8^4 = 2^{12}$ bins. Besides, some similar colors might be allocated to different nodes at the same level of the tree. Thus, a pruning process should be performed to reduce the complexity of the tree. We propose a four-step tree pruning process: initial node shrinking, node deletion, node merging and final node reduction. The detail of each step is discribed as following.

1. Step 1: Initial Node Shrinking
   Shrink the node whose pass-number is smaller than a threshold *t(l)*, where *l* is the level of a node located in the tree. This procedure is performed from top to down beginning with nodes at the first level of the tree. The reason of node shrinking is to simplify the complexity in Steps 3. The selection of threshold *t(l)* for deleting procedure influences the shape of the octree. In our experiment, we select *t(l)* to be 0.0001 of the total pass number, which means if there exist less than 0.01% pixels in the image process a certain color, this color can be ignored.
2. Step 2: Node Deleting
   Delete the nodes on the single child branch. The reason is that this kind of nodes do not provide extra color information than their parent. This procedure is performed based on a bottom-up fashion beginning with leaf nodes.
3. Step 3: Node Merging
   Cluster nodes with similar average colors at each level of the tree beginning with the leaf level. Then, merge ancestors at previous levels accordingly. The reason for node merging is to cluster similar colors beyond boundaries of fixed quantized subspaces.
4. Step 4: Final Node Reduction
   This procedure is similar to step 1 except it can be performed at the same time of feature output. The purpose of node reduction is to further simplify the octree feature.

Node merging is performed to overcome the problem of a fixed color quantization method where similar colors might be quantized to different bins of the histogram. We use a color clustering scheme to modify the octree which has a fixed quantization structure. The clustering is performed at each level of the tree. Two nodes in the color space with their average colors close to each other are added to the list of clustering nodes. In our implementation, two average colors are close to each other if they satisfy

$$\left|\vec{C}_{l,m} - \vec{C}_{l,n}\right| < d(l),$$

where $\vec{C}_{l,m}$ and $\vec{C}_{l,n}$ and are the average colors of nodes m and n at layer l of the tree, respectively. If nodes of similar colors are merged, their ancestors have to be adjusted accordingly. An illustration of the merging process is shown in Figure 4.4.

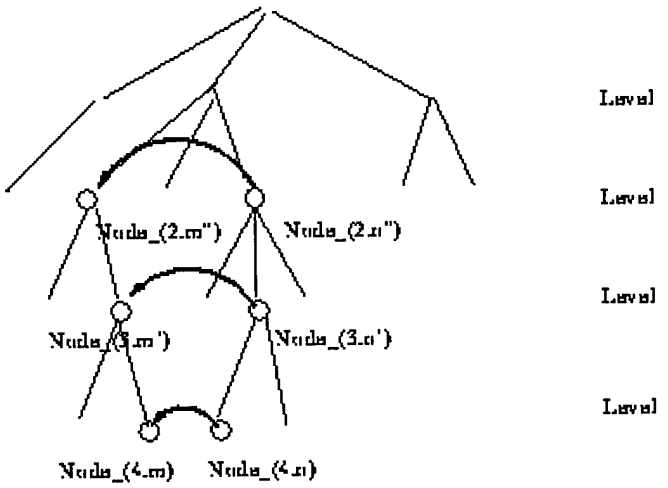

Figure 4.4 Illustration of the node merging process.

4.4 Octree in Linear Color Difference Spaces

4.4.1 Linear Color Difference Spaces

Many digital images are represented in RGB space, but the color difference computed using the Euclidean distance in RGB space is not totally consistent with the human visual system (HVS) model. That is, two colors with a large Euclidean distance in the RGB spaces may be perceptually similar. We study the construction of octree feature in other spaces in this section.

The quantitative measurement of the color distance has been studied extensively, and psychophysical experiments was conducted to determine the Just Noticeable Color Differences (JNCD). As we would expect, JNCD is not uniform along the three axes in the RGB space. The infinitesimal color difference *ds* of two neighboring colors can be written as

$$ds^2 = \sum_{i,j=i}^{3} C_{i,j} dx_i dx_j$$

where metric coefficients C_{ij} depend on x_i. To find the difference between two colors, we have to integrate the above equation from one color to the color. The integral is path-dependent and the actual distance is the integral along the path which yields the minimum distance between the two colors. Since the computational cost is high, an alternative approach is to map the RGB space onto another space with a uniform color difference. Several such spaces are CIEL*a*b*, CIEL*u*v* and Munsell color space [13]. The Munsell color space was named after artist Albert Munsell who created a book of colored samples ordered by the constant hue, brightness and saturation chart. The L*a*b* space was developed to provide a computationally simple measure of colors in agreement with the Munsell space. The L*u*v* space was evolved from the L*a*b* space and became the CIE standard in 1976. In Figure 4.5, we show the gamuts of the L*u*v* space translated from RGB space.

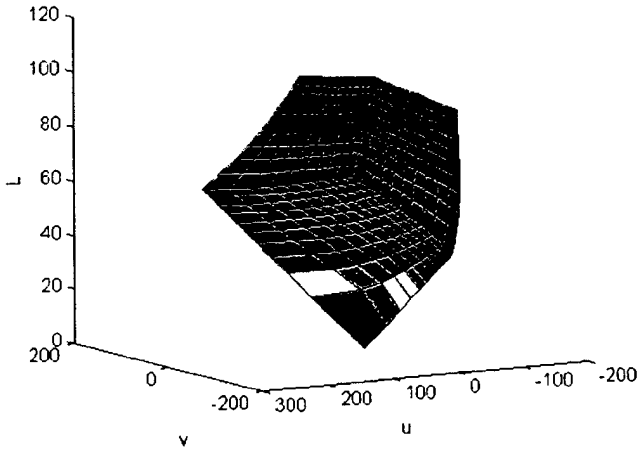

Figure 4.5 The gamuts of L*u*v* color space

4.4.2 Octree in CIE L*u*v* Space

It can be seen from Figure 5 that not all combinations of hue, chroma and value are within the gamut. Splitting the color space by a set of planes perpendicular to the axes is not very efficient

because a lot of subspaces are empty. Splitting schemes suitable to the irregular volumn of the L*u*v* space should be obtained. They do not have to seperate the L*u*v* space uniformly, because similar colors will be clustered beyond the boudary of subspaces eventually. One such set of splitting planes can be obtained by transforming the splitting planes in RGB space into L*u*v* space. Figure 6, demonstrates the splitting planes corresponding to the second level of the octree. Using this splitting scheme, we do not have to actually perform the transformation and initialize the octree in L*u*v* space by comparing the value of pixels with the boundaries defined by the slitting planes. When inserting a color into octree, we use the same bucket-finding method as in RGB space, which will lead to the same node. However, the average color of the node will be calculated in L*u*v* space, which will be useful in pruning process.

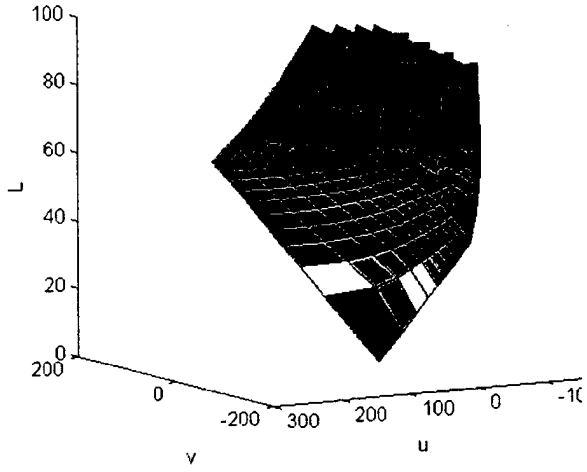

Figure 4.6 Splitting of the L*u*v* space.

The pruning process of octree in L*u*v* space also consists of four steps: initial node shrinking, node deletion, node merging and final node reduction. All steps are same as that in RGB space except the merging step. It is the merging step that resulted in a different octree in color feature representation. At this step, whether two nodes should be merged or not is determined by their color similarity in L*u*v* space. As introduced in the last subsection, the color distance in L*u*v* space is leanier to HVS. Thus the octree pruned in L*u*v* space will reveal the multiresolution color features relevant to HVS. To summarize, the procedure for octree construction in the L*u*v* space is:

1. Tree Initialization
   (a) Get the RGB Color of a new pixel in the image.
   (b) Find the belonging bucket using RGB components.
   (c) Update the pass-number.
   (d) Update the average color in L*u*v* space.
   (e) Go to the second step if it is not the last level of the tree.
   (f) Go to the first step if it is not the end of the image.
2. Tree Pruning
   (a) Shrink the node whose pass-number is less than a threshold to its parent.

(b) Delete the node without a brother from the bottom to the up.
(c) Merge the node in similar colors defined by the distance in L*u*v* space.
(d) Further reduce the node whose pass-number is less than a threshold to its parent.

4.5 Interactive Retrieval Using Octree

4.5.1 Relationship between Octree Shape and Image Color Appearance

We adopt the pruned octree as the indexing feature of an image by considering both its shape and content. The relationship between the color appearance of the query image and the shape of the pruned color quantization octree carries interesting information for image classification. Simply speaking, the width of the octree corresponds to richness of different colors of an image, while the depth of the tree corresponds to variation of similar colors in an image. For example, if the width of the octree is large, the image contains rich colors. One good example is the stained-glass image. The combined information of the width and depth of the tree provides us the color appearance of an image. Some typical examples are given in Table 4.1. Such an image classification step can be used to filter out irrelevant images effectively.

	Large width	Small width
Short depth	Normal images	Images with dominant colors
Long depth	Cartoon images	Company logos, traffic signs

Table 4.1 Categorization of images according to the octree structure.

4.5.2 Indexable Filtering Features of Octree

The following features are indexed to speed up the retrieval based on the shape and content of the pruned octree.

- Average color:

The average color attribute of the root node is the average color of the entire image. The distance of images Q and T by the average color is defined False alarms occur if only this feature is used in filtering. But, it is needed in the sequential filtering stage at the end.

- Dominant color:

If the query image has a dominant color, then the pass-number of one node will be much larger than that of the other nodes at the first level. The average color of the node is the dominant color for that image. The distance defined by dominant color is: By comparing only the dominant color, a large amount of irrelevant images can be deleted from the candidate list.

- Color depth:

The depth of the octree is related to the color depth of an image. Images with the same shape but a with different color depth are visually different. A sketch picture of a landscape looks very different with a photograph of the same scenery. The color depth of the later is much larger. Filtering by the color depth is helpful in getting images with different color flavors.

- Color width:

The maximum width of the octree is related to color richness of an image so that it is also useful in discriminating irrelevant images.

- Tree shape:

Similar images should have a similar octree shape. To compare the shapes of two trees, a distance is defined as the sum of common nodes between image Q and T.

- Layered color distributions

The average color and the pass-number of nodes at each level define a set of multiresolution color distributions. These distributions can be used for sequential filtering in the query process. The layered distance is defined as the sum of the pass-number of the common nodes for image Q and T.

4.5.3 Query Examples Based on Octree Features

Filtering by a partial set of octree features can be performed at the beginning stage of image retrieval to exclude irrelevant images if the query image has a certain prominent features, e.g. a certain dominant color, an unusual color width or color depth. Sequential filtering based on layered comparison of the octree can be performed at a later stage to refine the candidate similar image set. We use several examples to demonstrate this idea. Our image database consists of more than 2,100 images, including natural scenes, animals, plants, architectures and people. Large varieties of our image database prevent the bias for a particular type of images. Three image sets, i.e. "Skiing", "Stained-glasses", and "sunset" are used as query image sets. A typical image of each set is shown in Figure 4.7.

Figure 4.7 Typical images in the query set.

Retrieval of "Skiing" image

Each image in the "Skiing" image set is dominated by the white tone. The dominant color and the percentage of pixels possess the color is shown in Table 4.2. Retrieval by the dominant color alone in this case can promptly get the candidate image set.

	Position	Dominant color	Pass-Number
Ski_0	Node_{0,7	(179,194,199)	0.769694
Ski_1	Node_{0,7	(234,232,200)	0.908407
Ski_2	Node_{0,7	(212,214,197)	0.811035
Ski_3	Node_{0,7	(205,214,191)	0.899821
Ski_4	Node_{0,7	(203,204,179)	0.859456
Ski_5	Node_{0,7	(249,242,216)	0.892415
Ski_6	Node_{0,7	(185,190,179)	0.816691
Ski_7	Node_{0,7	(179,200,212)	0.713704
Ski_8	Node_{0,7	(215,211,196)	0.854533
Ski_9	Node_{0,7	(187,186,171)	0.726847

Table 4.2 The dominant color of images in "Skiing" set.

Retrieval of "Stainedglasses" image

The color width of the query image is large. The statistic of the width of the octree with respect to the entire database at the first level is shown in Table 4.3. It can be seen that only 0.66% images have a width greater than 7. Filtering by the color width helps to narrow down the set of candidate images quickly. The scheme of filtering by the color width is suitable for two types of images, i.e. images with very rich colors or a limited number of distinct colors.

Color Width	1	2	3	4	5	6	7	8
No. of images (%)	1.27	19.53	45.21	22.56	8.73	2.08	0.47	0.19

Table 4.3 The color width of images in the database.

Retrieval of "Sunset" image

Each image in this set has a dominant color, where the dominant color might not be very similar. For example, some images are dominated by the dark red, while other images are dominant in the dark yellow color. Thus, the threshold for filtering by the dominant color has to be set a larger value to avoid false misses and, as a result, the candidate image set becomes larger. Filtering by the layered color distribution can be applied to this set of candidate images. In Table 4.4, we show the lowest rank of the image in the query set (the size of minimum candidate set) over the number of images of the entire database after each step of filtering.

Filtering Step	Tree-level 1	Tree-level 2	Tree-level 3	Tree-level4
Rank	7.64%	5.23%	3.21%	1.23%

Table 4.4 The size of candidate images at various levels after sequential filtering

4.6 Conclusion

We have explored a octree-based color feature for image indexing and retrieval. The discriminating power of the new color feature is better than the multiresolution histogram proposed in our previous work because it reflects the color clustering in each individual image. We have also explored the new feature construction method in the linear color distance space such as L*u*v* to further improve the retrieval performance. The new color feature provides rich indexing features such as the dominant color, the color depth, the color width and the multiresolution color distributions. A combination of these features not only supports a very flexible query process but also speeds up the retrieval process.

4.7 References

[1] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, "Query by image and video content: The qbic system," *IEEE Computer* , vol. 28, pp. 23--32, September 1995.

[2] H. Chen, B. Schatz, T.Ng, J. Martinez, A. Kirchhoff, and C. Lin, "A parallel computing approach to creating engineering concept spaces for semantic retrieval: the illinois digital library initiative project," *IEEE trans. on Pattern Recognition and Machine Intelligence* , vol. 18, no. 8, pp. 771--782, 1996.

[3] M. Swain and D. Ballard, "Color indexing," *International Journal of Computer Vision* , vol. 7, no. 1, pp. 11--32, 1991.

[4] H. Zhang, C. L. Y. Gong, and S. Smolia, "Image retrieval based on color features: an evaluation study," in *SPIE Digital Image Storage and Archiving Systems* , vol. SPIE 2606, pp. 212--220, Oct 1995.

[5] F. Liu and R. Picard, "Periodicity, directionality and randomness: Wold features for image modeling and retrieval," Tech. Rep. Technical Report No. 320, MIT Media Laboratory and Modeling Group, 1994.

[6] B. Manjunath and W. Ma, "Texture features for browsing and retrieval of image data," *IEEE trans. on Pattern Recognition and Machine Intelligence* , vol. 18, no. 8, pp. 837--842, 1996.

[7] R. Mehrotra and J. Gary, "Similar-shape retrieval in shape data management," *IEEE Computer*, vol. 28, pp. 57--62, September 1995.

[8] M. Stricker and M. Orengo, "Similarity of color images," in *SPIE Storage and Retrieval for Image and Video Databases III*, vol. SPIE 2185, pp. 381--392, Feb. 1995.

[9] X. Wan and C.-C. Kuo, "Color distribution analysis and quantization for image retrieval," in *SPIE Storage and Retrieval for Image and Video Databases IV*, vol. SPIE 2670, pp. 9--16, Feb 1996.

[10] X. Wan and C.-C. Kuo, "Image retrieval with multiresolution color space quantization," in *Electronic Imaging and Multimedia Systems* , November 1996.

[11] J. Hafner, H. Sawhney, W. Equitz, M. Flickner, and W. Niblack, "Efficient color histogram indexing for quadratic form distance functions," *IEEE trans. on Pattern Recognition and Machine Intelligence*, vol. 17, pp. 729--736, July 1995.

[12] M. Stricker, "Color indexing with weak spatial constraints," in *SPIE Storage and Retrieavl for Image and Video Databases IV*, vol. 2670, pp. 29--40, Feb. 1996.

[13] G. Wyszecki and W.S. Stiles, *Color Science*. New York, John Wiley Sons, 1982.

[14] M. Gervautz and W. Purgtathofer, *A Simple Method for Color Quantization: Octree Quantization*. San Diego: Academic Press, 1990.